ISSN 2464–4617 (print) ISSN 2464–4625 (CD-ROM)

CSRN 2703

(Eds.)

Vaclav Skala
University of West Bohemia, Czech Republic

Computer Science Research Notes

25. International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision
WSCG 2017
Plzen, Czech Republic
May 29 – June 2, 2017

Proceedings

WSCG 2017

Posters Proceedings

ISSN 2464–4617 (print) ISSN 2464–4625 (CD-ROM)

ISSN 2464–4617 (print) ISSN 2464–4625 (CD-ROM)

CSRN 2703

(Eds.)

Vaclav Skala
University of West Bohemia, Czech Republic

Computer Science Research Notes

25. International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision
WSCG 2017
Plzen, Czech Republic
May 29 – June 2, 2017

Proceedings

WSCG 2017

Posters Proceedings

Vaclav Skala – Union Agency

ISSN 2464–4617 (print) ISSN 2464–4625 (CD-ROM)

Computer Science Research Notes
CSRN 2703

Editor-in-Chief: Vaclav Skala
c/o University of West Bohemia
Univerzitni 8
CZ 306 14 Plzen
Czech Republic
skala@kiv.zcu.cz *http://www.VaclavSkala.eu*

Managing Editor: Vaclav Skala

Publisher & Author Service Department & Distribution:
Vaclav Skala - UNION Agency
Na Mazinach 9
CZ 322 00 Plzen
Czech Republic
Reg.No. (ICO) 416 82 459

ISSN 2464-4617 (Print) ***ISSN 2464-4625*** (CD/DVD)
ISBN 978-80-86943-51-0 *(Print)* ***ISBN 978-80-86943-46-6*** *(CD/DVD)*

WSCG 2017

International Program Committee

Adzhiev, Valery (United Kingdom)
Anderson, Maciel (Brazil)
Benes, Bedrich (United States)
Bilbao, Javier,J. (Spain)
Bourke, Paul (Australia)
Daniel, Marc (France)
de Geus, Klaus (Brazil)
Drechsler, Klaus (Germany)
Feito, Francisco (Spain)
Ferguson, Stuart (United Kingdom)
Galo, Mauricio (Brazil)
Giannini, Franca (Italy)
Gobbetti, Enrico (Italy)
Gudukbay, Ugur (Turkey)
Juan, M.-Carmen (Spain)
Kenny, Erleben (Denmark)
Kim, Jinman (Australia)
Kim, HyungSeok (Korea)
Lobachev, Oleg (Germany)
Molla, Ramon (Spain)
Montrucchio, Bartolomeo (Italy)
Muller, Heinrich (Germany)
Murtagh, Fionn (United Kingdom)
Pan, Rongjiang (China)
Pedrini, Helio (Brazil)
Platis, Nikos (Greece)
Ramires Fernandes, Antonio (Portugal)
Richardson, John (United States)
Ritter, Marcel (Austria)
Rojas-Sola, Jose Ignacio (Spain)
Sanna, Andrea (Italy)
Segura, Rafael (Spain)
Skala, Vaclav (Czech Republic)
Sousa, A.Augusto (Portugal)
Szecsi, Laszlo (Hungary)
Teschner, Matthias (Germany)
Tokuta, Alade (United States)
Umetani, Nobuyuki (Japan)
Wu, Shin-Ting (Brazil)
Wuensche, Burkhard,C. (New Zealand)
Wuethrich, Charles (Germany)
Yao, Junfeng (China)

WSCG 2017

Board of Reviewers

Aburumman, Nadine (France)
Adzhiev, Valery (United Kingdom)
Anderson, Maciel (Brazil)
Assarsson, Ulf (Sweden)
Averkiou, Melinos (Cyprus)
Ayala, Dolors (Spain)
Benes, Bedrich (United States)
Bilbao, Javier,J. (Spain)
Capobianco, Antonio (France)
Carmo, Maria Beatriz (Portugal)
Charalambous, Panayiotis (Cyprus)
Cline, David (United States)
Daniel, Marc (France)
Daniels, Karen (United States)
de Geus, Klaus (Brazil)
De Martino, Jose Mario (Brazil)
de Souza Paiva, Jose Gustavo (Brazil)
Diehl, Alexandra (Germany)
Dokken, Tor (Norway)
Dong, Yue (China)
Drechsler, Klaus (Germany)
Eisemann, Martin (Germany)
Feito, Francisco (Spain)
Ferguson, Stuart (United Kingdom)
Galo, Mauricio (Brazil)
Garcia-Alonso, Alejandro (Spain)
Gdawiec, Krzysztof (Poland)
Giannini, Franca (Italy)
Gobbetti, Enrico (Italy)
Gobron, Stephane (Switzerland)
Goncalves, Alexandrino (Portugal)
Gonzalez, Pascual (Spain)
Gudukbay, Ugur (Turkey)
Hernandez, Benjamin (United States)
Jones, Mark (United Kingdom)
Juan, M.-Carmen (Spain)
Kenny, Erleben (Denmark)
Kim, HyungSeok (Korea)
Kim, Jinman (Australia)
Kurillo, Gregorij (United States)
Kurt, Murat (Turkey)
Lee, Jong Kwan Jake (United States)
Liu, Yang (China)
Liu, Beibei (United States)
Liu, SG (China)
Liu, Damon Shing-Min (Taiwan)
Lobachev, Oleg (Germany)
Luo, Shengzhou (Ireland)
Marques, Ricardo (Spain)
Mei, Gang (China)
Mellado, Nicolas (France)
Meng, Weiliang (China)
Mestre, Daniel,R. (France)
Meyer, Alexandre (France)
Molina Masso, Jose Pascual (Spain)
Molla, Ramon (Spain)
Montrucchio, Bartolomeo (Italy)
Muller, Heinrich (Germany)
Murtagh, Fionn (United Kingdom)
Nishio, Koji (Japan)
Oberweger, Markus (Austria)
Oyarzun Laura, Cristina (Germany)
Pan, Rongjiang (China)
Pedrini, Helio (Brazil)
Pereira, Joao Madeiras (Portugal)
Pina, Jose Luis (Spain)
Platis, Nikos (Greece)
Puig, Anna (Spain)
Raidou, Renata Georgia (Austria)
Ramires Fernandes, Antonio (Portugal)

Richardson, John (United States)
Ritter, Marcel (Austria)
Rodrigues, Joao (Portugal)
Rojas-Sola, Jose Ignacio (Spain)
Sanna, Andrea (Italy)
Schwaerzler, Michael (Austria)
Segura, Rafael (Spain)
Serano, Ana (Spain)
Sik-Lanyi, Cecilia (Hungary)
Sommer, Bjorn (Germany)
Sousa, A.Augusto (Portugal)
Szecsi, Laszlo (Hungary)
Teschner, Matthias (Germany)
Todt, Eduardo (Brazil)
Tokuta, Alade (United States)
Tytkowski, Krzysztof (Poland)
Umetani, Nobuyuki ()
Umlauf, Georg (Germany)
Vanderhaeghe, David (France)
Vidal, Vincent (France)
Vierjahn, Tom (Germany)
Wu, Shin-Ting (Brazil)
Wuensche, Burkhard,C. (New Zealand)
Wuethrich, Charles (Germany)
Yao, Junfeng (China)
Yoshizawa, Shin (Japan)
YU, Qizhi (United Kingdom)
Zhao, Qiang (China)

WSCG 2017

Poster Papers Proceedings

Contents

Ray Tracing API Integration for OpenGL Applications

Wei-Hao Lai
Industrial Technology Research Institute
195, Sec. 4, Chung Hsing Rd., Chutung, Hsinchu, 31040, Taiwan
WeiHaoLai@itri.org.tw

Chang-Yu Tang
Industrial Technology Research Institute
195, Sec. 4, Chung Hsing Rd., Chutung, Hsinchu, 31040, Taiwan
CYTang@itri.org.tw

Chun-Fa Chang
National Taiwan Normal University
162, Sec. 1, Heping E. Rd., Taipei City 106, Taiwan
chunfa@ntnu.edu.com

ABSTRACT

Ray tracing is one of the most important rendering techniques in computer graphics. By means of simulating reflection and refraction of light transportation, ray tracing generates more photorealistic images than scanline rendering. However, the high computational cost is the main disadvantage of ray tracing algorithm. In recent years, the computing power of GPU has increased dramatically, and general-purpose computing on graphics processing units (GPGPU) has become popular. Many scholars have presented some physically based rendering methods with CUDA or OpenCL in order to improve image quality and increase rendering speed. Because rasterization is the mainstream in the gaming industry, there is still a long way to go to make ray tracing accepted by the industry in the near future. We introduce a ray tracing API integration for OpenGL applications that can replace the original OpenGL rasterization with ray tracing by simply adding a few lines of code, and the ray tracing algorithm in this API is parallelized by OpenCL.

Keywords

3D Rendering, Ray Tracing, Game Engine, Parallel Computing, OpenCL

1. INTRODUCTION

Ray tracing is widely used in the special effects and 3D animation industry. These commercial products emphasize photorealistic scenes and high quality lighting effects. Scenes and animations are rendered offline, so the speed is not the primary concern. However, due to the boom of 3D game industry, customers nowadays are not only asking for vivid scenes, but also instant interactions. Using general-purpose computing on graphics processing units (GPGPU) architecture, ray tracing now can be parallelized by GPU to make real-time rendering possible.

Rasterization is the main image synthesis method in the 3D game industry (e.g., OpenGL, Direct3D, and Vulkan). Beside the consideration of speed, the reason why it might be difficult to replace rasterization with ray tracing is that developers need to be familiar with the new structure of API to alter the original source code. In this paper, we propose an OpenGL-like API to substitute the OpenGL native rasterization to ray tracing, helping developers achieve the global illumination effects with minimal modifications.

OpenGL 3.0 introduced a deprecation mechanism to simplify future revisions of the API [Shr09a]. The direct-mode rendering using *glBegin* and *glEnd* function calls is one of the deprecated features. Vertex buffer object (VBO) is considered to be a more efficient way to make draw calls instead. VBO allows vertex array data to be stored in high-performance graphics memory on the server side and promotes efficient data transfer. We will show a few OpenGL applications with our ray tracing API integration to demonstrate its feasibility.

The ray tracing API integration is separated into two parts. The first part is collecting parameters from OpenGL API calls, redirecting each call to our API and capture parameters from original OpenGL source code at the same time. The other part is the ray tracing program written in OpenCL based on C99.

Due to the innate limitation, ray tracing algorithm starts when the whole scene data is ready. Once the ray tracing kernel is called, the API returns the rendered frame back to OpenGL and displays the frame on screen by the render-to-texture method. In other words, OpenGL would not rasterize the frame, but only display the frame rendered by our ray tracing API.

The paper is organized as follows. Section 2 provides an overview of the related work. In section 3, we describe the implementation and structure of our ray tracing API integration. More details about ray tracing

algorithm implementation are shown in section 4. Some experiment results and comparison between our method and the original OpenGL are in section 5. Finally, we conclude in section 6.

2. Related Work

The PowerVR OpenRL is a flexible low level API for accelerating ray tracing in both graphics and non-graphics applications [Img17a]. By writing OpenRL shading language (RLSL), developers can obtain full control of ray tracing logic. There are three types of shaders in OpenRL. Vertex shader handles vertex positions in the scene. Frame shader sets rays and sampling methods. A ray shader defines the ray color and the energy that a ray contains.

The OptiX is an application framework for achieving optimal ray tracing performance on the GPU [Nvi17a]. OptiX provides a simple, recursive, and flexible pipeline for accelerating ray tracing algorithms. Many professional 3D computer graphics software are using it as a plug-in (e.g., AutoCAD, Maya, and Lightworks). Though Optix is a powerful and mature framework, it only works on Nvidia graphic cards. Considering OpenGL cross-platform feature, our ray tracing API integration has to minimize dependency as far as possible.

Brian Paul started the Mesa project in 1993 [Pau17a], which is an open-source implementation of the OpenGL specification. Mesa implements a translation layer between a graphics API such as OpenGL and the graphics hardware drivers in the operating system kernel. Our method is quite similar to Mesa. The difference is instead of simulating the OpenGL pipeline, we use ray tracing instead.

OpenCL is the open standard for cross-platform and parallel programming of diverse processors found in personal computers, servers, mobile devices, and embedded platforms [Khr17a]. An OpenCL program is divided to run on host and device. The host is the main CPU used to configure kernel execution. The device is the component that contains the processing units that will execute the kernel. A host can trigger multiple kernels to do diverse missions, and assign to different devices.

3. Ray Tracing API Integration

The ray tracing API integration has two parts. The first part is called the function calls redirection, which is responsible for parameters collection. When OpenGL API works, the parameters are packed and redirected to our API for later use. The second part is a ray tracer written in OpenCL, which undertakes the parameters from the previous step and passes them to a kernel program. After receiving the parameters from the host, the kernel program implements ray tracing in parallel.

In this paper, the ray tracing API integration is easy to use. Programmers only need to include a header file, and then an OpenGL application would automatically render with ray tracing, but without the original rasterization.

3.1 Supported OpenGL Versions

According to OpenGL 3.0 specification, the fixed function pipeline as well as most of the related OpenGL functions and constants were declared deprecated. These deprecated elements and concepts were commonly referred to legacy OpenGL [Seg17a]. How to distinguish a legacy OpenGL source code is not a difficult task. One typical sign that a program is using Legacy OpenGL is immediate mode. Immediate mode is using *glBegin* and *glEnd* with *glVertex* and *glColor* in between them

The advantages of legacy OpenGL are built-in lighting model, procedural-base method, etc. GPU hardware architecture improves in leaps and bounds. The fixed function pipeline is unable to match the flexibility. The OpenGL Shading Language (GLSL) has been added to allow for increasing flexibility of the rendering pipeline at the vertex and fragment level. Nevertheless, there are still some OpenGL applications use the immediate mode, such as education courses and tutorial websites because it is easier to learn.

We concentrate on applications that use client-side VBO, and give those applications the ray tracing integration support, but we do not support immediate mode starts with *glBegin* and ends with *glEnd*. Due to unified shading architecture, we do not know how the vertex attribute pointers are used in GLSL. The applications containing GLSL program would be skipped by our ray tracing integration API. However, modern OpenGL programs must contain at least one GLSL program. The problem is solved by creating a hint module added at the front of every vertex attribute pointer to tell our API what those VBOs actually do, and then read GLSL programs, trying to interpret by our API. The idea is inspired by the OpenRL programming model [Img17a].

3.2 API structure

Figure 1 shows the ray tracing API integration flow chart. As we can see on the leftmost side is a common OpenGL application flow containing model data loading, light adjusting, data pointer definition, draw calls, and synchronization. At the middle is what our API does to the OpenGL API. These OpenGL API calls would not present their original behaviors, but execute data collection and transfer data to our specification. The rightmost is the ray tracing flow chart. We start up a ray tracing kernel when the program meets the synchronization point. A space-partitioning data structure called KD-tree is used to boost the ray-triangle intersection test in the algorithm. The test scenes in our experiments are static because every time an object moves, the tree must be rebuilt

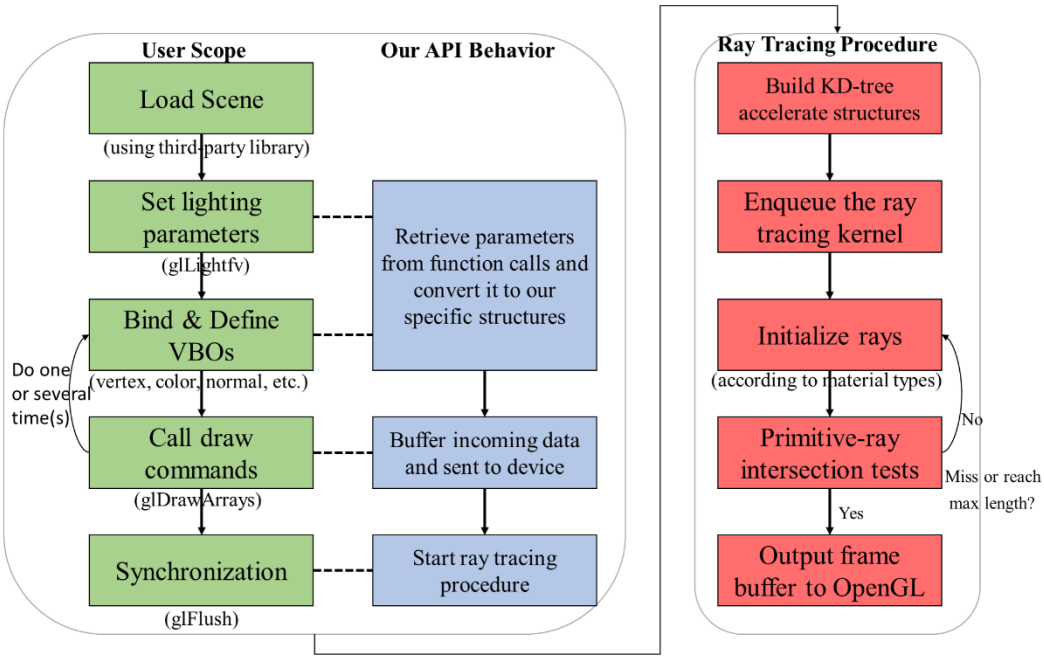


Figure 1: Ray tracing API integration flow chart. The leftmost green flow is user scope, and it is about the execution sequence of an OpenGL application. The middle is about capturing parameters from original OpenGL functions. The rightmost is the ray tracing procedure written in OpenCL.

and our implementation would not be able to reach real-time performance. Danilewski, et al. [Dan10a] constructed a binned SAH KD-tree with CUDA to solve the dynamic scene problem and declared their method could reach real-time, but now we just focus on our method that works on static scenes.

3.3 Supported Function Calls

Legacy OpenGL applications use built-in matrix manipulation functions to change the position of 3D objects, fixed-function lighting model to add effects, and client-side VBO to upload data. These actions are done during the data preparation stage in our ray tracing API integration. The ray tracing kernel in our API would wait until synchronization call signal. Table 1 shows all supported function calls, which are enough for a simple OpenGL application execute smoothly. Chapter 5 shows some results rendered by our ray tracing API and detail analysis would be discussed later.

Buffer	Rendering	Capability
glGenBuffers	glVertexPointer	glEnableClientState
glBindBuffer	glColorPointer	glDisableClientState
glBufferData	glNormalPointer	glEnable
	glDrawArrays	glDisable
	glFlush	
Lighting	**GLU(OpenGL Utility Library)**	**Extension**
glLightfv	gluPerspective	rtMaterialEXT
	gluLookAt	rtBuildAcclStructEXT

Table 1: Supported function calls. The extension column represents some features that legacy OpenGL does not support.

4. Ray Tracing

In this section, we will talk about the rendering algorithm in our ray tracing API. The rendering method is a ray tracing program developed with OpenCL.

4.1 Overview

Ray tracing describes a method for producing visual images constructed in 3D computer graphics environments [Whi05a]. To implement ray tracing algorithm, the steps are as follows. First, construct vectors from observer to each sample on the image plane. These vectors are deemed as lights in the space. Second, calculate intersections with all 3D geometrics in the scene and compute the color using lighting models. Then, the third step is to generate new rays according to reflection models and refraction models.

4.2 Implementation Details

Back to Figure 1, rightmost: we build an acceleration data structure called KD-tree. This step helps us conserve some unnecessary intersection calculations and improve rendering speed. Our ray tracing kernel is inspired by Whitted ray tracer model [Whi05a], but there are some differences in detail. In OpenCL, we create a kernel that execute in parallel, though the model is not. The second point is that the original model expresses in recursive style. However, OpenCL does not support recursion. Consequently, it is necessary to convert recursion to iteration-based implementation.

The intersection test is one of the most time-consuming computation in ray tracing algorithm. A reputable intersection computation method gains more computational efficiency. The Möller-Trumbore algorithm is a fast ray-triangle intersection algorithm [Mol05a]. It calculates the intersection of a ray and a triangle in three dimensional space without needing precomputation of the plane equation of the plane that contains the triangle. In our implementation, triangle and sphere object types are supported.

We apply the diffuse part of Phong shading [Pho75a] of illumination when intersection points are calculated. Some other physical effects such as reflection and refraction are also implemented in our API. It is worth notice that while using our ray tracing integration API, we can add these effects in an OpenGL application program though the original source code does not have these effects. The most common used accelerating structure of ray tracing algorithm are BVH-tree and KD-tree. We wrote a KD-tree for our API, and we refer to Pharr et al [Pha04a]. The KD-tree is a binary tree in which every node is a k-dimensional point. Every non-leaf node can be thought of as implicitly generating a splitting hyperplane that divides the space into two parts. While doing a ray intersection test, traverse through a KD-tree can prune half of the triangles that are not hit, preventing the unnecessary intersection being computed.

5. RESULTS

This section presents the results obtained by our ray tracing API integration. The experiment environment is according to Table 2. The image resolution is set to 800*600.

Specifications	
OS	Windows 10 pro x64
CPU	Intel i7-6700K
RAM	DDR4 2666 32GB
NVIDIA® GeForce GTX TITAN X	
CUDA Cores	3072
Memory	12 GB G5
TFLOPS	7

Table 2: Experiment environment

Figure 2 are images rendered by the original OpenGL rasterization and our ray tracing API integration. The left pictures are rendered by OpenGL and the right ones are rendered by our API. The picture at the bottom-right has a hard shadow effect that is undoubtedly easy to ray tracing, but OpenGL would need a bunch of GLSL code to achieve the same result. Figure 3 shows more results rendered by our API. The top-left is a Cornell box with a reflective red wall and a dragon in it. The top-right is a glass cube and a dragon in a Cornell box. The paired images at the bottom is a dragon and Sponza separately. Table 3 shows the frame per second (FPS) of each scene. B. & D. represents Cornell box and dragon. The "w/ Reflec." and "w/ Refrac." mean "with reflection" and "with refraction". These features only affect our API integration, so OpenGL column has no test results. Although our ray tracing API is still slower than rasterization. However, it is worth to be mentioned that in most test cases, the FPS data of our API are above 30 FPS, which is the minimum threshold of real-time rendering. This means that the hardware is now able to bear such large amount of computing throughput.

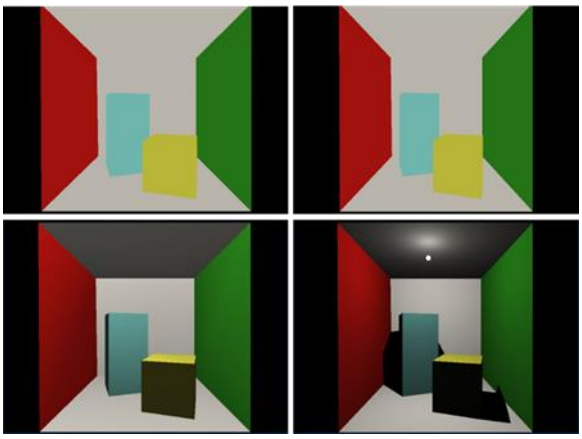


Figure 2: The comparison of rendering results between original OpenGL (left) and our ray tracing API integration (right).

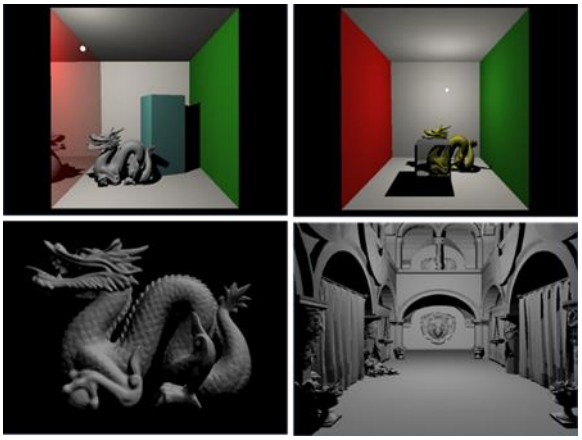


Figure 3: More experiments by proposed method.

	# Tri.	**OpenGL**	**Our API**
Cornell Box	34	5730.58 FPS	261.72 FPS
B. & D.	100,034	5149.60 FPS	53.62 FPS
B. & D. w/ Reflec.	100,034	---------	50.40 FPS
B. & D. w/ Refrac.	100,034	---------	49.30 FPS
Sponza	262,267	2612.49 FPS	14.39 FPS

Table 3: FPS comparison between OpenGL and our API

	KD-tree build time (second)	Triangle intersections per frame	Sphere intersections per frame	Avg. Intersections per second
Cornell Box	0.000044	4113244	480000	17550.2216
B. & D.	0.503594	3805833	480000	79929.7464
B. & D. w/ Reflec.	0.503594	4868135	539321	107290.7937
B. & D. w/ Refrac.	0.503594	5125656	501409	114139.2495
Sponza	1.512600	15000425	480000	1075776.5810

Table 4: Detail information about ray intersections

Typically the intersection test is a bottleneck of the ray tracing algorithm. Except the construction of KD-tree, we make the rest actions all done by GPU to benefit from its parallelization. According to Table 3 and Table 4, if the average intersections per second is around a hundred thousand, the FPS can be 50 or even more. While rendering the Sponza scene, the average intersections come up to a million, but the FPS still remains around 14.

6. Conclusion

The contribution of this paper is the introduction of a ray tracing API integration for OpenGL applications. Our API makes it much easier to develop a ray tracing based application by using OpenGL code at hand. Moreover, a ray tracing based application can reach interactive speed is in practice. Our API still have a lot of room for improvement. For example, we can use Monte Carlo path tracing techniques mentioned by Keller et al. [Kel12a]. Another improving direction is to break the limit of static scenes. Danilewski et al. [Dan10a] constructed SAH KD-tree on GPU with CUDA, which could achieve real-time rendering of dynamic scenes.

7. Acknowledgement

This work is supported in part by the Ministry of Science and Technology (Taiwan) grant MOST 103-2221-E-003-010-MY3, the Industrial Technology Research Institute grant D0-10404-28-3, and by the Institute for Information Industry.

8. REFERENCES

[Dan10a] Danilewski, P., Popov, S., & Slusallek, P. Binned sah kd-tree construction on a gpu. Saarland University, 1-15.

[Img17a] Imagination T. OpenRL SDK - Imagination Community. Retrieved January 4, 2017, from https://community.imgtec.com/developers/powervr/openrl-sdk/

[Kel12a] Keller, A., Premoze, S., & Raab, M. Advanced (Quasi) Monte Carlo Methods for Image Synthesis. In ACM SIGGRAPH 2012 Courses.

[Khr17a] Khronos Group. OpenCL - The open standard for parallel programming of heterogeneous systems. Retrieved January 4, 2017, from https://www.khronos.org/opencl/

[Pau17a] Paul, B. Mesa Introduction. Retrieved January 4, 2017, from http://www.mesa3d.org/

[Mol05a] Möller, T., & Trumbore, B. Fast, minimum storage ray/triangle intersection. In ACM SIGGRAPH 2005 Courses (p. 7). ACM.

[Nvi17a] NVIDIA C. NVIDIA OptiX Ray Tracing Engine | NVIDIA Developer. Retrieved January 4, 2017, from https://developer.nvidia.com/optix

[Pha04a] Pharr, M., & Humphreys, G. Physically based rendering: *From theory to implementation. Morgan Kaufmann*.

[Pho75a] Phong, B.T. Illumination for computer generated pictures. Communications of the ACM, 18(6), 311-317.

[Seg17a] Segal, M., & Akeley, K. The OpenGL Graphics System: A Specification (Version 3.0 - September 23, 2008). Retrieved January 4, 2017, from https://www.opengl.org/registry/doc/glspec30.20080923.pdf

[Shr09a] Shreiner, D., & Bill The Khronos OpenGL ARB Working Group. OpenGL programming guide: the official guide to learning OpenGL, versions 3.0 and 3.1. Pearson Education, 2009.

[Whi05a] Whitted, T. An improved illumination model for shaded display. In ACM Siggraph 2005 Courses (p. 4). ACM.

EVALUATION OF STANDARD AND DIRECTIONAL MEDIAN FILTERS ALGORITHMS AND THEIR IMPLEMENTATION ON FPGA FOR MEDICAL APPLICATION

Talbi, F.; Alim-Ferhat, F.; Seddiki, S.; Hachemi, B.
Centre de Développement des Technologies Avancées
BP 17, Baba Hassen, Algiers 16303, Algeria
falim@cdta.dz

ABSTRACT

The standard and the directional median filters are effective methods for the removal of impulse-based noise from the images. The main advantage is being the preserving of edges as compared to the mean filter.
The main objective of this article is to implement the standard and the directional median filters on FPGA (Field Programmable Gate Array) in order to eliminate impulsive noise in the medical image. As a first step, two algorithms were developed and validated by Matlab tool.
Subsequently, two architectures are proposed and implemented using the Xilinx ISE 12.2 environment.

Index Terms: Impulsive noise, Median filter, Directional median filter, PSNR, FPGA.

1. INTRODUCTION

THE world of science has used in several image processing technique to improve image quality; indeed, this latter is often affected by noise during acquisition. The impulsive noise is one of the major problems in medical image processing. Impulse noise from digital images can be removed by using mean, standard median and directional filters. Impulsive noise is commonly appears during images acquisition; to eliminate it, median filters are used [1, 2].

A reliable and precise filtering of normal or pathological medical images from imaging systems remains an important objective in processing of medical information. This is because it represents the first step in the analysis chain leading to assisted diagnosis in studying internal structures morphology.

In the literature, there are many kinds of filters realizing this task. However, some of them have disadvantages such as degradation of the medical image quality, suppression of elements containing helpful information in the diagnosis, etc.

To increase the quality and precise filtering, standard and directional median filters are used in many industrial applications that go from machine vision to robotics going through medical imaging.

The standard and directional median filters are used widely in medical image processing since it preserves edges while removing noise.

Our objective is the elimination of impulsive noise in medical images. When noise intensity is increasing, a standard median or directional filter remains many shots unfiltered. To overcome the limitation of standard and directional median filters, some works have been proposed in the literature to improve the performances and the efficiently of these filters. Several works have been realized; among them, implementation based on FPGA [3, 4, 5]. To achieve our goal, we have organized this paper as follows.

The first section introduces filters and their various applications. The second section deals with the operating principle of the two filters studied in this paper, and their implementation in MATLAB software tool. The third section focuses on hardware implementation on FPGA programmable circuit, simulation results and temporal performances. The last section concludes the paper and draws-up perspectives and future works on the development of directional median filter.

2. STANDARD MEDIAN FILTER

The median filter operates according to the following steps [6,7] :

1) Sort the values in ascending order.
2) Replace the value of the central pixel by the value in the middle of the sorted values
3) Repeat this process for all image pixels.

Generally, the sorting is done in ascending order. The ordered elements were noted f_i, the ascending sort is characterized by:

$$f_1 < f_2 < \ldots < \frac{f_{n+1}}{2} < \ldots < f_{n-1} < f_n$$

The median element of the neighborhood is $\frac{f_{n+1}}{2}$.

2.1 Software implementation

To verify the efficiency and quality of the proposed filtering technique, we test this procedure on several MRI (Magnetic Resonance Images) images using MATLAB tool. The program was applied to medical images with different noise levels.

Fig. 2 below illustrates the application of the standard median filter on noisy MRI images with a rate of 20% with an application window of 3x3 filter.

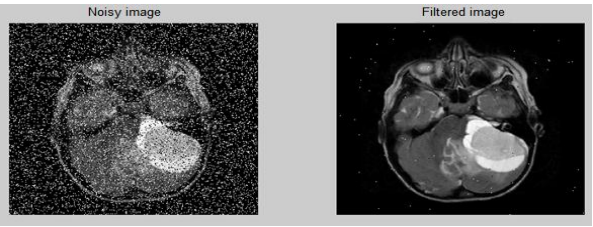


Figure 1 Application of the median filter on MRI images by MATLAB tool for the image

3. DIRECTIONAL MEDIAN FILTER

This filter uses a pulse detector, which is based on the differences between the current pixel and its neighbors aligned with four main directions.

After the pulse detection, it does not just replace the noisy pixels identified by median filter outputs but continues to use the information of the four directions to weigh the pixels in the window to preserve the details.

3.1 Method of operation

In this section, we describe the method of operation of the directional median filter following next steps [8]:

Step 1: Pulse detector

Fig. 2 shows the edges aligned with the four main directions

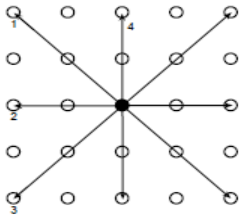


Figure 2 Graphical representations of the four directions

According to the previous figure, each of these four directions contains 05 pixels. S_k (k=1 .. 4) represents a set of coordinates aligned with the k^{th} direction centered at (0, 0).

In the 5x5 window centered at (i, j), we calculate the index $d_{i,j}^{(k)}$ for each direction; Knowing that $d_{i,j}^{(k)}$ is the absolute sum of all the differences in gray levels values between $y_{i+s,j+t}$ and $y_{i,j}$ with (s, t)ϵS_k^0 ,$d_{i,j}^{(k)}$ is the direction index defined by:

$$d_{i,j}^{(k)} = \sum_{(s,t)\,\epsilon S_k^0} \left| y_{i+s,j+t} - y_{i,j} \right| \quad 1 \le k \le 4$$

Each index direction is sensitive to edge aligned with a given direction. Then, the minimum of these four directions indices is used for pulse detection, which can be denoted by:

$r_{i,j} = \min\left\{ d_{i,j}^{(k)} : 1 \le k \le 4 \right\}$ The value of r(i, j) can be discussed in 3 cases:

- When the current pixel is a flat region without noise, $\boldsymbol{r_{i,j}}$, is small, because the four indices direction are small.
- When the current pixel is an edge pixel, $\boldsymbol{r_{i,j}}$ is also small, because at least one indice direction index is small.
- When the current pixel is a pulse, $\boldsymbol{r_{i,j}}$ is great, because the four indices directions are great.

Then we establish a threshold T to compare the coefficient $\boldsymbol{r_{i,j}}$ in order to determine if the pixel is noisy or not. The pulse detector is defined as follows:

Then we establish a threshold T to compare the coefficient $\boldsymbol{r_{i,j}}$ in order to determine if the pixel is noisy or not.

The pulse detector is defined as follows:

$$y_{i,j} = \begin{cases} \text{noisy pixel} & \text{if } r_{i,j} > T \\ \text{pixel noiseless} & \text{if } r_{i,j} \le T \end{cases}$$

Step 2: Calculation of the median value

After the pulse detection, the noisy pixels will be replaced by the calculated median values according to the direction which is the lowest standard deviation.

The first step is to calculate the standard deviation $\sigma_{i,j}^{(k)}$ of gray levels values between $y_{i+s,j+t}$ and (s, t) ϵS_k^0 (k=1 à 4). The equation of $L_{i,j}$ is given by:

$$L_{i,j} = \operatorname{argmin}\left\{ \sigma_{i,j}^{(k)} : k = 1 \text{ to } 4 \right\}$$

With:

$$\sigma_{i,j}^{(k)} = \sqrt{1/5 \sum_{(s,t)\,\epsilon S_k^0} (y_{i+s,j+t} - \text{mean}^{(k)})^2]}$$

$$\text{mean}^{(k)} = \text{mean}\left(\text{the pixel values in the } k^{\text{th}} \text{direction}\right)$$

The argmin operation is used to calculate the function minimizer. The standard deviation shows how values are grouped in the average value of all pixels.

$L_{i,j}$ Indicates that the pixels are on the same direction as the central pixels are closer to each other. So to reconstruct the noisy pixel, the following equation is used:

$$m_{i,j} = \text{median}\left\{ W_1 * y_{i+s,j+t} : (s,t)\epsilon\Omega^3 \right\}$$

$$\text{With } W_1 = \begin{cases} 2, & (s,t)\epsilon S_{li,j}^0 \\ 1, & \text{elsewhere} \end{cases}$$

According to the equations listed above, the output of the directional median filter is given by:

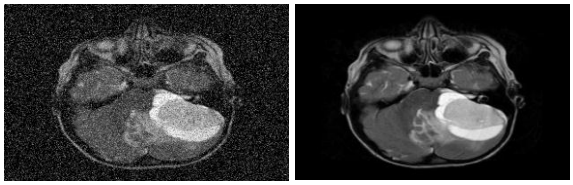
$$u_{i,j} = \alpha_{i,j} y_{i,j} + \left(1 - \alpha_{i,j}\right) m_{i,j}$$

With

$$\alpha_{i,j} = \begin{cases} 0 & \text{si } r_{i,j} > T \\ 1 & \text{si } r_{i,j} \le T \end{cases}$$

3.2 Software implementation:

Fig. 3 below shows the result of the application of directional median filter on noisy MRI images with a noise rate of 20% under the MATLAB tool.



(a1) (a2)

Figure 3 (a1) Medical image with 20% impulsive noise, (a2) filtered image with the directional median filter

3.3 Comparison of the two filters PSNR parameter

According to the results of the calculation of the PSNR for the median filter and the directional median, we make a comparison in order to justify our choice for directional median filter [9].

$$\text{EQM} = \frac{\sum(M(i,j) - T(i,j))^2}{m * n}$$

$$\text{PSNR} = 10 * \log\frac{255^2}{\text{EQM}}$$

Fig. 4 s shows the different values of the PSNR for the two filters:

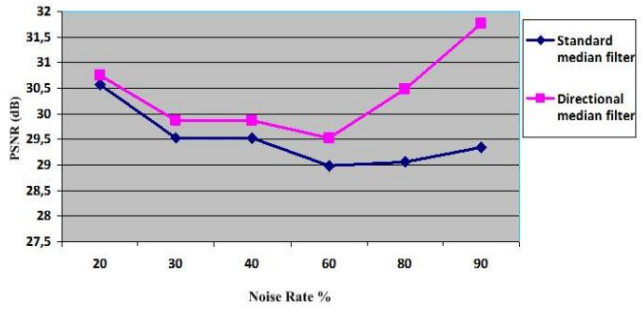


Figure 4 Variation of the value of the PSNR for both filters

Results discussion

Indeed, the above graph shows that the difference range between the two values in the same image and the same noise rate begins to widen even more once the noise rate reaches or exceeds 60%.

4. HARDWARE IMPLEMENTATION OF THE STANDARD AND THE DIRECTIONAL MEDIAN FILTER

In this part, we present the architecture, the simulation and the implementation of both median filters: Standard and directional median filters.

4.1 Description of the global architecture of the standard median filter

The global architecture of the standard median filter is composed of two different blocks which are:

1) Memory block.
2) Sort block.

The architecture of the memory block is represented by Fig.5:

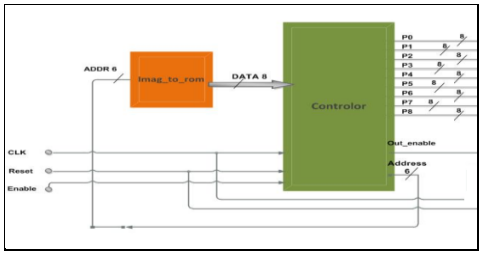


Figure 5 Architecture of the memory block

The memory block is constituted by:

1) RAM (Random Access Memory).
2) A controller.

The Sort block is realized by a set of comparators. A 3x3 sliding window algorithm is used as the base for filter operation, the sort is accomplished by using 36 comparators, according the following equation:

$$N = c_{ws}^2$$

C: combination; *N*: comparators numbers; *ws*: window size.

A comparison is made between two values (8 bits) to define the higher value and the lower value as shown in Figure 6.

At the output of the block, the nine pixels are sorted in descending order. So the value of the central pixel (P4) of the 3x3 matrix must be replaced by the value in the middle of the sorted values (Median). This operation is repeated for all pixels of the image.

Finally, the architecture which has been developed of the standard median filter is shown in Fig. 6:

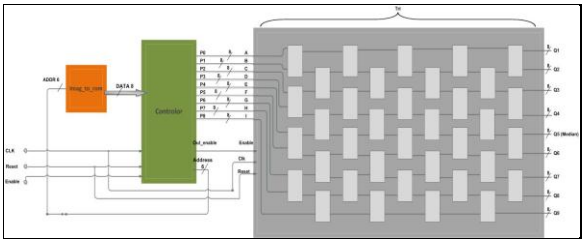


Figure 6 Standard median filter developed architecture

4.2 Description of the global architecture of the directional median filter

The block diagram of the filter is composed of five blocks defined as follows:

1) Block memory.
2) Calculation block of the directional indices and the impulse detector.
3) Calculation block of a standard deviation.
4) Calculation block of minimizer.
5) Calculation block of the median value.

Fig. 7 describes the global proposed architecture:

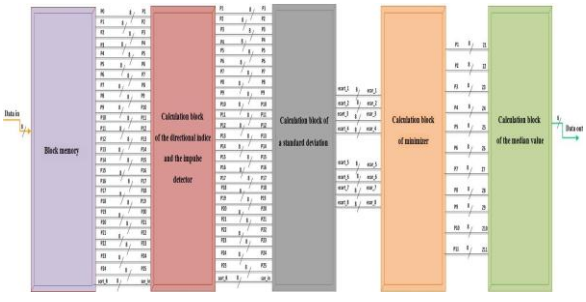


Figure 7 Block diagram of the directional median filter

The memory block is as the same as the previously described block; but, the operation to generate the pixels must be repeated 25 times to have at the output 25 pixels.

In the calculation block of the directional index and the

impulse detector, we need 25 pixels as the block input (Direction); so, a 5x5 matrix to calculate the 4 indices direction outputs, these outputs are introduced at the input of the block (TRI), this block is realized by 6 comparators, according to the above equation. Each comparator must do a comparison between two directions indices to select the smallest direction which is the output that represents the value of the pulse detector, this value must be compared with a threshold to determine if the pixel is noisy or not.

The calculation block of a standard deviation calculates the standard deviation in four directions; so, we need 25 pixels at the input of block **(sous_ecart)**. This block is realized by this equation:

$$\boldsymbol{sust} = [\mathbf{1/5} \sum_{(s,t)\ \epsilon S_k^0} (y_{i+s,j+t} - \boldsymbol{mean}^{(k)})^2]$$

According to architecture below (Figure 8), **Sust_i** realize the direction i, (i= 1 to 4).

The second part is the block (**Squart_root**), this latter used core generator based on CORDIC algorithm which calculates the square root of the equation above.

The minimizer block determines the direction that represents the smallest standard deviation. When this direction is found we will take a matrix of 3X3 size centered on the pixel (i, j) and performs a duplicate operation of the two pixels of the selected direction.

In the calculation block of the median value, the sorting is accomplished through a number of comparators 5x5. Each comparator must make a comparison between two 8-bit values to define the maximum value and the minimum value.

At the end of this operation the eleven pixels are sorted in descending order. So the value of the central pixel of the 5X5 matrix must be replaced by the value located in the middle of the sorted values, and this operation is repeated for all pixels of the image.

Fig. 8 shows the architecture of a standard deviation:

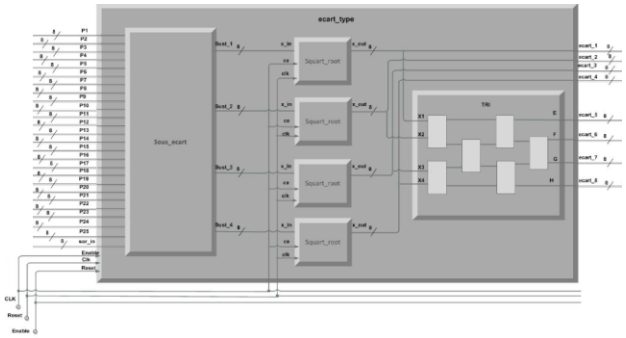


Figure 8 Standard deviation block

4.3 Comparison resources of the two architectures

The resources performances of the two architectures are shown in table.1:

Table 1 Comparison resources of the two median filters

	Number of Slice Registers	Number of Slice LUTs	Number used as logic	Number of occupied Slices	Number of bonded IOBs	Frequency (MHZ)	Execution times (s)
Median standard	514 (1%)	481 (1%)	481 (1%)	227 (3%)	83 (17%)	368.216	$1.755.10^{-7}$
Directional median	1.397 (4%)	1.563 (5%)	1.551 (5%)	601 (8%)	91 (18%)	107.202	$2.274.10^{-7}$

From the above comparisons, we notice that the standard median filter uses less execution time and few resources with satisfactory performance; however, directional median filter is preferable in terms of precision; despite the number of resources which is very large.

5. CONCLUSION

Image filtering is still a very large area of research. The objective of our work is devoted to the filtering of cerebral medical images from MRI « Magnetic Resonance Images ».

The image filtering is the heart of many problems in medical imagery because often it constitutes the first step of a real image processing flow of image depending on application selected.

We have implemented both architectures (i) standard and (ii) directional median filter under ISE tool of XILINX.

First step is devoted to the software implementation of these filters, under MATLAB tool, applied to MRI noisy. The results achieved at the simulation show the efficiency of the directional median filter algorithm.

The second step, we have presented the methodology of hardware design of the architectures developed of the standard and the directional median filters. The C5VLX50T 3FF1136-circuit of VIRTEX-5 family used for hardware implementation: Performances were justified in terms of the hardware resources used, and the execution time.

In order to integrate this work in a processing chain such as medical image segmentation, we must improve in the future:

The optimizing of the memory block to have fast access to the pixels stored in RAM. The other improvement consists of increasing of the calculation accuracy.

6. REFERENCES

[1] Sadri AR, Zekri M, Sadri S, Gheissari N, “Impulse noise cancellation of medical images using wavelet networks and median filters”, J Med Signals Sens. 2012 Jan;2(1):25-37.

[2] Toprak A, Güler I, “Suppression of impulse noise in medical images with the use of Fuzzy Adaptive Median Filter”, J Med Syst. 2006 Dec;30(6):465-71

[3] Sarala Singh, Prof. Ravimohan,"A VHDL Implementation of Effective Impulse Noise Filtration Technique", International Journal of Computer Technology and Electronics Engineering (IJCTEE), Volume 3, Issue 3, June 2013.

[4] Manju Chouhan, C.D Khare, "Implementation of Median Filter for CI Based on FPGA", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 12, December 2013, ISSN: 2278 – 1323.

[5] S.A. Fahmy, P.Y.K. Cheung W. Luk3, "High-throughput one-dimensional median and weighted median filters on FPGA", Published in IET Computers & Digital Techniques, ISSN 1751-8601.
[6] Hanifi Djamel, "Générateur d'architectures pour plate-forme reconfigurable dédié au traitement d'image", Thèse de Doctorat, Université d'Evry Val d'Essonne, Evry, France, 2000.
[7] Chikh Mohammed Tahar, "Amélioration des images par un modèle de réseau de neurones (Comparaison avec les filtres de base) ", thèse de Master en Informatique, 28 Septembre 2011.
[8] J.K. Mandal, AparnaSarkar, "A Modified Weighted Based Filter for Removal of Random Impulse Noise (MWB)", Second International Conference on Emerging Applications of Information Technology, 2011.
[9] J. K. Mondal, Somnath Mukhopadhyay" New Directional Weighted Median Filter for Removal of Random-Valued Impulse Noise", ICCS – 2010, November 19-20, 2010.

VIDEO SUMMARIZATION BASED ON LOCAL FEATURES

Mohamed Massaoudi Sahbi Bahroun Ezzeddine Zagrouba

Research Team on Intelligent Systems in Imaging and Artificial Vision (SIIVA) - LIMTIC Laboratory
Institut Supérieur d'Informatique (ISI), Université Tunis El-Manar
2 Rue Abou Rayhane Bayrouni, 2080 Ariana, Tunisia

med.massaoudi@yahoo.fr Sahbi.Bahroun@isi.rnu.tn ezzeddine.zagrouba@fsm.rnu.tn

ABSTRACT

Keyframe extraction process consists on presenting an abstract of the entire video with the most representative frames. It is one of the basic procedures relating to video retrieval and summary. This paper present a novel method for keyframe extraction based on SURF local features. First, we select a group of candidate frames from a video shot using a leap extraction technique. Then, SURF is used to detect and describe local features on the candidate frames. After that, we analyzed those features to eliminate near duplicate keyframes, helping to keep a compact set, using FLANN method. We developed a comparative study to evaluate our method with three state of the art approaches based on local features. The results show that our method overcomes those approaches.

Keywords

Video Summarization, Keyframe Extraction, Interest Points, SURF, FLANN.

1. INTRODUCTION

Videos have turned out to be the main source of information, learning and entertainment, with the growing advancements and progress in multimedia technologies. Daily, millions of videos are being uploaded on Internet consisting of news, sports clips, tutorials, lectures contents and many more. Content based retrieval of video has emerged as a growing challenge and therefore, automatic keyframes extraction; the main step for the efficient retrieval, video classification and story retrieval; has become so important and vital.

Using one keyframe as shot representation was considered by the majority of works found on the literature which defined, for example, as the shot first frame or median frame. Nevertheless just one frame, in most cases, is not able of fully representing the variety of information in a shot, usually composed by hundreds of images that can have different content [1].

Therefore, in this work, we propose a novel method for keyframe extraction based on local features. Due to their capabilities of retaining image semantics and providing robust descriptors, local features were the most reliable and widely applied method in the image retrieval field [2-5]. However, they have been poorly explored in the video keyframe extraction domain.

The rest of this paper is organized as follows. In Section 2 we briefly introduce three state of the art keyframe extraction approaches based on local features found in literature [6, 7, 8]. We will describes the proposed keyframe extraction method in section 3. Section 4 presents and analyzes the experimental results obtained. Finally, Section 5 concludes the paper and opens some perspectives of future work.

2. RELATED WORKS

In order to represent a shot, compact approaches are usually adopted which most of them are based on keyframes and color histograms. Those methods suffers a low representativeness and supposes that the video is already segmented into shots by a shot detection algorithm. Many works in the literature have been proposed for keyframes extraction based on local features. Furthermore, we discuss three keyframe extraction methods based on local features found in the literature.

Baber et al. [6] used SURF features to describe each extracted keyframe. First, the video is segmented into shots, then; the keyframes are defined as the shot median frame for each shot. Even if this approach has low computational cost, since it considers only a small fraction of the available frames, there is the issue of selecting an image, which is the median frame in this case, that does not represent the most relevant content of the shot.

Chergui et al. [7] consider that a relevant image contains rich visual details. Thus, they defined the keyframe as the frame with the highest number of points of interest in the shot. Despite using images content, it is not possible to guarantee that the frame with the highest number of points of interest is the most representative one in all cases. Besides, one image may not be enough to describe the diverse content

of some shots and important information can be lost. This method is also more computationally demanding, because the selection step involves processing all shot frames.

Tapu and Zaharia [8] extract a variable number of frames from each shot using a leap extraction technique. Then, each frame is compared with the existing keyframes already extracted. If the visual dissimilarity between them is significant, the frame is added to the keyframes set. After that, the extracted keyframes are described by SIFT. This method may have the advantage that not all shot frames are processed, but many parameters need to be set what can influence the quality of the shot representation.

The related work presented in this section show that the use of local features can be a substitute for keyframe representations. However, as discussed, the current approaches present problems of representativeness and computational costs leading to high processing times

3. PROPOSED APPROACH

We developed a keyframe extraction method based on local features, designed to deal with the problems identified in related work and discussed on at section 2, i.e., representativeness and computational cost due to high processing times.

3.1. Candidates Frames Selection

In order to select the best frames to be the keyframes of each shot, we, initially, select some frames into a Candidates Set (CS). The first frame to be included in the CS is defined as the shot first frame. This has the goal of guarantee that each shot will be represented by, at least, one keyframe. The next frames to be included in the CS follow a windowing rule. We defined a window of size n and the frames at positions n+1, 2n+1, 3n+1, and so on, are selected for later analysis. We set the fps value for n because within 1 second there is no significant variation on consecutive frames content.

Algorithm 1 Candidates frame generation

Require: Video V={$f_1, f_2, \ldots, f_n$}
1: fps = V.getFPS()
2: i = 1
3: while i < n do
4: cs.add(f_i)
5: i = i + fps
6: end while
7: return cs

3.2. Keyframes extraction

The next step is to extract SURF [9] features from the frames in the CS. The result is a number of feature vectors, of 64 dimensions, representing each frame. SURF features matching is faster compared to other descriptors such as SIFT. SURF features are also invariant to scale, rotation and partial illumination change [10]. The exact number of vectors varies according to the frames content but it is generally high. This is another reason to adopt the windowing rule (mentioned before) instead of to use all frames in the shot (see Figure 1).

Algorithm 2 Keyframes extraction

Require: Candidates Set $cs=\{cf_1, cf_2, \ldots, cf_m\}$
1: keyframes.add(cf_1)
2: i = 2
3: while i < m
4: U = extractSURF(cf_i)
5: isKeyFrame = True
6: for k = 1 to ks.size() do
7: V = extractSURF(cf_k)
8: M = matching(U,V)
9: y = $(1 - \frac{M}{|U|}) \times 100$
10: if y < 80% then
11: isKeyFrame = False
12: end if
13: end for
14: if isKeyFrame = True then
15: keyframes.add(cf_i)
16: end if
17: i = i + 1
18: end while
19: return keyframess

The first keyframe extracted is the first one in the CS. Then, each frame in the CS is analyzed according to the following criterion: it will be considered as a keyframe only if it has more than 80% (which is according to the literature the typical value used in vision applications) [11] of feature vectors different from each keyframe already extracted. The matching score y is defined as:

$$y = (1 - \frac{M}{|U|}) \times 100 \quad (1)$$

where M is number of matched features, |U| is number of features in the analyzed candidate frame. The reasoning behind this criterion is to avoid the extraction of similar keyframes, since similar keyframes do not add value to representativeness. We used the FLANN method proposed in [12] for automatically selecting the best matching method and its parameters for a given training set. It was shown to be fast in practice and is part of the OpenCV library.

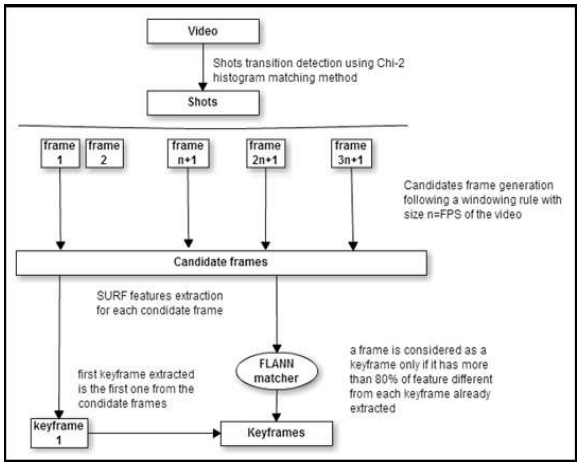


Figure 1. Proposed method steps.

3.3. FLANN matcher

FLANN stands for Fast Library for Approximate Nearest Neighbors which is a library of feature matching methods. It can provide automatic selection of index tree and parameter based on the user's optimization preference on a particular data-set. Automatic algorithm configuration allows a user to achieve high performance in approximate nearest neighbor matching by calling a single library routine. The user need only provide an example of the type of dataset that will be used and the desired precision, and may optionally specify the importance of minimizing memory or build time rather than just search time.

4. EXPERIMENTAL RESULTS

The efficiency of the proposed keyframe extraction method was evaluated by experimental tests on some videos (news, cartoons, games,…). These video present different challenges (camera motion, background-foreground similar appearance, dynamic background,…). Results proved that the method can extract efficiently keyframes resuming the salient semantic content of a video with no redundancy.

To verify the effectiveness of the proposed method, we first use qualitative evaluation since the subjective evaluation of the extracted key frame is efficient and it was used in many state of the art methods. In a second step, we will complete the evaluation with a quantitative study by calculating fidelity and compression rate. The use of quantitative and qualitative evaluation can prove the effectiveness of our proposed approach. The experiments were done on movies from YUV Video Sequences (http://trace.eas.asu.edu/yuv/) and some other standard test videos with different sizes and contents. In this paper we will show experiments done only on 7 movies as example. These movies were already segmented into shots by the $\chi 2$ histogram matching method [13]. Table 1 shows the number of frames and shots for the seven movies.

Movie	Frames	Shots
News	300	4
Bus	150	6
Foreman	297	3
Mother and Daughter	300	1
Suzie	150	4
Salesman	449	8
Carphone	382	7

Table 1. The videos characteristics

4.1. Validity Measures

For validity measures we used the fidelity and the compression rate.

4.1.1 Fidelity

The fidelity measure is based on semiHausdorff distance to compare each key frame in the summary with the other frames in the video sequence. Let $V_{seq}=\{F_1,F_2,\ldots F_N\}$ the frames of the input video sequence and let KF all keyframes extracted $KF=\{F_{K1}, F_{K2},\ldots, F_{KM}\ldots\}$. The distance between the set of key frames and F belonging to V_{seq} is defined as follows:

$$DIST(F,KF) = Min\{Diff(F,F_{Kj})\}_{,j=1\ to\ M} \quad (2)$$

Diff() is a suitable frame difference. This difference is calculated from their histograms: a combination of color histogram intersection and edge histogram based dissimilarity measure [14] .The distance between the set of key frames KF and the video sequence V_{seq} is defined as follows:

$$DIST(V_{seq},KF) = Max\{Diff(F_i,\mathrm{KF})\}_{,i=1\ to\ N} \quad (3)$$

So we can define the fidelity (FD) as follows:

$$FD(V_{seq},KF) = MaxDiff - DIST(V_{seq},KF) \quad (4)$$

MaxDiff is the largest value that can take the difference between two frames Diff (). High Fidelity values indicate that the result of extracted keyframes from the video sequence provides good and global description of the visual content of the sequence.

4.1.2 Compression Rate

Keyframe extraction result should not contain many key frames in order to avoid redundancy. That's why we should evaluate the compactness of the summary. The compression ratio is computed by dividing the number of key frames in the summary by the length of video sequence. For a given video sequence, the compression rate is computed as follows:

$$CR = 1 - \frac{card\{keyframes\}}{card\{frames\}} \quad (5)$$

Where card{keyframes} is the number of extracted key frames from the video. Card{frames} is the number of frames is the video.

4.2. Qualitative Evaluation

Now, we will present some results for 2 examples of videos. The first one is "news.mpg" which has 300 frames segmented into 4 shots. The figure 3 shows the 2 resulting key frames. As we can see the first image in figure 3 is the keyframe relative to the first shot of "news" video presented in figure 2 which is very logic, furthermore, the redundancy was eliminated.

The second video is "foreman.mpg" it is composed of 297 frames and segmented into 3 shots. The figure 5 shows the resulting keyframes for all the video. In the same way the first image of figure 5 is the keyframe relative to the first shot of "foreman" video (figure 4). Table 2 summarizes the number of key frames extracted for each video.

Movie	Number Of keyframes
News	2
Bus	4
Foreman	3
Mother and Daughter	1
Suzie	2
Salesman	3
Carphone	3

Table 2. Number of keyframes extracted per video

4.3. Quantitative Evaluation

We measured now for each movie, the fidelity and the compression rate (CR %). The table 3 illustrates these results.

Movie	Fidelity	CR%
News	0.80	99.33
Bus	0.69	97.33
Foreman	0.74	98.90
Mother and Daughter	0.77	99.60
Suzie	0.81	98.60
Salesman	0.77	99.32
Carphone	0.80	99.21

Table 3. Results in term of fidelity and compression rate

While looking to the results in Table 3 by the compression ratio (CR) values, it is clear that the proposed method minimizes considerably the redundancy of the extracted keyframes which guarantees encouraging compression ratios while maintaining minimum requirements of memory space. The Fidelity values confirm the same interpretation that we get by looking to the compression rate.

In order to give an objective evaluation, we compared the resulting quality measures of compression rate of our proposed method with three state of the art methods [6, 7, 8] and this for the seven tested videos in Table 3.

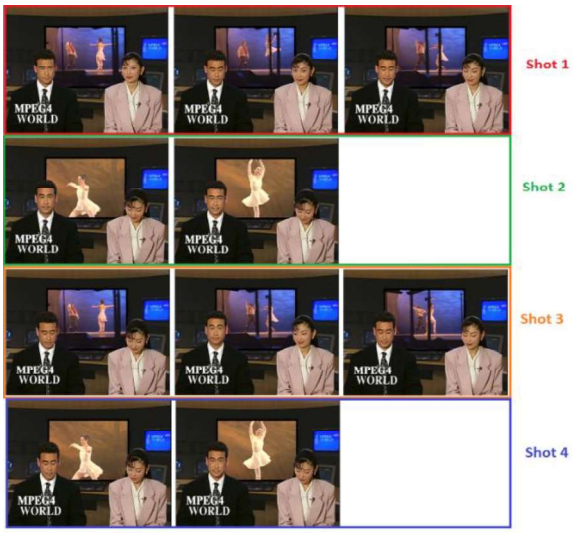


Figure 2. Shots of the video "news.mpg".

Figure 3. Keyframes of the video "news.mpg".

Figure 4. Shots of the video "foreman.mpg".

Figure 5. Keyframes of the video "foreman.mpg".

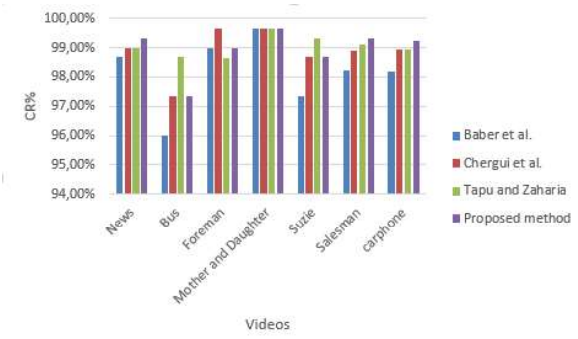


Figure 6: Comparison of the quality of the extracted keyframes in term of compression rate (CR%).

In Figure 6, we show a comparison between our proposed method and three state of the art methods in terms of compression rate. As the CR value is high as we have different keyframes. We can see in Figure 6 that our proposed method reduced considerably the redundancy of extracted keyframes.

5. CONCLUSIONS

In this paper, we have proposed a simple and effective technique for keyframe extraction based on SURF local features and using the FLANN matching method. Firstly, candidate frames are selected adaptively using a leap extraction method. Each candidate frame is described by SURF local features vectors. Secondly, keyframes for each shot are selected from the candidate frames set using FLANN method to discard any duplicated keyframes. The proposed approach proved to have superior effectiveness to three state of the art related work, i.e., gives a set of image that covers all significant events in the video while minimizing information redundancy in keyframes.

As a perspective, we consider developing a complete system for still image-based face based on visual summary which is composed by faces from the extracted keyframes. The user can initiate his visual query by selecting one face and the system respond with videos which contains that face.

6. REFERENCES

[1] Souza, T.T. and Goularte, R. 2013. Video Shot Representation Based on Histograms. Proceedings of the 28th ACM Symposium on Applied Computing (Coimbra, Portugal, 2013), 961–966.

[2] Baber, J., Satoh, S., Afzulpurkar, N. and Keatmanee, C. 2013. Bag of Visual Words Model for Videos Segmentation into Scenes. Proceedings of the Fifth International Conference on Internet Multimedia Computing and Service (New York, NY, USA, 2013), 191–194.

[3] Blanken, H.M., Vries, A.P., Blok, H.E. and Feng, L. 2010. Multimedia Retrieval. Springer.

[4] Lowe, D.G. 2004. Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision. 60, 2 (2004), 91–110.

[5] Lowe, D.G. 1999. Object recognition from local scale-invariant features. Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on (Kerkyra, Greece, 1999), 1150 – 1157 vol.2.

[6] Baber, J., Afzulpurkar, N. and Bakhtyar, M. 2011. Video segmentation into scenes using entropy and SURF. Emerging Technologies (ICET), 2011 7th International Conference on (2011), 1–6.

[7] Chergui, A., Bekkhoucha, A. and Sabbar, W. 2012. Video scene segmentation using the shot transition detection by local characterization of the points of interest. Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on (2012), 404–411.

[8] Tapu, R. and Zaharia, T. 2011. A complete framework for temporal video segmentation. Consumer Electronics – Berlin (ICCE-Berlin), 2011 IEEE International Conference on (2011), 156–160.

[9] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," Comput. Vis. Image Underst., vol. 110, no. 3, pp. 346– 359, 2008.

[10] S. Bahroun, H. Gharbi and E. Zagrouba, "Local query on satellite images based on interest points," 2014 IEEE Geoscience and Remote Sensing Symposium, Quebec City, QC, 2014, pp. 4508-4511.

[11] Dror Aiger, Efi Kokiopoulou, Ehud Rivlin. Random Grids: Fast Approximate Nearest Neighbors and Range Searching for Image Search. The IEEE International Conference on Computer Vision (ICCV), 2013, pp. 3471-3478

[12] M. Muja, D. G. Lowe, Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration, in International Conference on Computer Vision Theory and Applications (VISAPP'09), 2009

[13] Cai and al, 2005. A Study of Video Scenes Clustering Based on Shot Key Frames . Series Core Journal Of Wuhan University (English) Wuhan University Journal Of Natural Sciences Pages 966-970

[14] Ciocca, G., and Schettini, R. 2006. An innovative algorithm for key frame extraction in video summarization. J. of Real-Time Image Processing 1(1): 69-88.

A new dimensionality reduction-based visualization approach for massive data

Kotryna Paulauskienė
Institute of Mathematics and Informatics,
Vilnius University
Akademijos str. 4
LT-08663 Vilnius, Lithuania
kotryna.paulauskiene@mii.vu.lt

Olga Kurasova
Institute of Mathematics and Informatics,
Vilnius University
Akademijos str. 4
LT-08663 Vilnius, Lithuania
olga.kurasova@mii.vu.lt

ABSTRACT

We live in a big data and data analytics era. The volume, velocity, and variety of data generated today require special methods and techniques for data analysis and inferencing. Data visualization tools allow us to understand the data deeper. One of the straightforward ways of multidimensional data visualization is based on dimensionality reduction and illustrated by a scatter plot. However, visualization of millions of points in a scatter plot does not make a sense. Usually, data sampling or clustering is performed before visualization to reduce the amount of the visualized points, but in such a case, meaningful outliers can be rejected and will not be visualized. In this paper, a new approach for massive data visualization without point overlapping is proposed and investigated. The approach consists of two main stages: selection of a data subset and its visualization without overlapping. The experiments have been carried out with ten data sets. The efficiency of subset selection and visualization of data subset projection is confirmed by a comprehensive set of comparisons.

Keywords

Massive data, dimensionality reduction, data visualization, data subset, visualization without overlapping.

1. INTRODUCTION

Today, big data handling is still challenging. The big data analysis helps us to do insights that lead to better decisions and strategic moves. The analysis of large amounts of data helps us to reveal hidden patterns, correlations, and other insights. Usually, the analysed real-world data is of high-dimensionality. In general, each data instance (point) is characterized by many features. In order to visualize data on a 2D or 3D space, the dimensionality needs to be reduced to two or three. Dimensionality reduction (projection) techniques extract lower-dimensional data from the high-dimensional input data [1]. The techniques map the data points from m-dimensional space to a smaller d-dimensional one ($d < m$). 2D and 3D spaces are used for data visualization. Data visualization approaches present data in a graphical form and enable people to understand the data better and deeper. Good data visualization yields better models and predictions and allows discovering the unexpected information [2], [3]. The size of data, obtained and generated, at present is huge and continues to increase every day [4]. Usually, big data is characterized by three main components: volume, velocity, and variety [5], [6]. In this paper, high-dimensional and large volume data is considered to be massive data. Visualization of a large amount of data points in a scatter plot in most cases will end with some shape, fully filled with data points, and it won't be an informative representation of the data. Human eyes have a difficulty in extracting meaningful information when the data becomes extremely large. Not many existing visualization systems are designed to present meaningful and high-quality information for human perception of big data [7]. Another problem we face is that some dimensionality reduction techniques cannot handle a large amount of data. The multidimensional scaling algorithm is usually unable to deal with large amount of points and requires much computational time [8], [9]. Thus, in this research, we suggest to visualize not a full data set but only a subset of the data. Having only a subset of high-dimensional points, the projection can be found fast and then visualized in a scatter plot [10]. Also, there are many open source and commercial data visualization tools, but in most of them dimensionality reduction methods are not implemented, visualization can be performed by two or three features or visualization is applied to aggregated (counted, averaged, min/max values) data [11], [12], [13] The following most common types of graphs can be mentioned: scatter plot, line chart, bar

chart, pie chart, histogram, infographic, and others [14]. Thus, we need a visualization approach that helps us to comprehend a large amount of data and to identify each data point location among the data.

2. VISUALIZATION APPROACH

The proposed visualization approach consists of two main stages: (1) proper selection of a data subset and calculation of the data subset projection; (2) visualization of projection of the data subset without point overlapping.

Consider a data set $X \subset R^m$ with n points. Let $X_S = \{x_1, \dots, x_s\} \subset X, s < n$, be a subset of the data set, for which a set of corresponding low-dimensional points $Y_S = \{y_1, \dots, y_s\} \subset R^d, s < n, d = 2$ is computed using any dimensionality reduction method. The final data subset for visualization is $Y_L = \{y'_1, \dots, y'_l\} \subset Y_S, l \leq s < n$. The final subset for visualizing Y_L contains less data points than Y_S, since we eliminate the overlapping points. The input in the proposed visualization approach is high-dimensional points of the given data and the obtained output is the data subset points of reduced dimensionality to be visualized.

1.1. Selection of a data subset

An important task of the proposed strategy is a proper selection a data subset. A conventional way is to cluster the data and then select representatives from each cluster, but in this case, we can lose the outliers. Data clustering is one of the most popular techniques in data mining. It is a process of partitioning an unlabelled data set into clusters, where each cluster contains data points that are similar to one to another with respect to a certain similarity measure and different from that of other clusters [15], [16]. For data subset selection the following methods can be used: simple random sampling, systematic sampling, stratified sampling, cluster sampling, etc. [17]. But in this case, not all outlying observations that can be significant in data analysis and knowledge discovery are selected. For example, only one or two outliers will be selected to the data subset because they usually are far from other observations and their density is not high. An outlier can be defined as an observation that is far distant from the rest of observations [18]. An outlier may be due to variability in the measurement or it can indicate an experimental error, but also it can be due to a chance or some natural process of the construct that is being measured. Outliers are often considered as an error or noise, however, sometimes they can carry important information about the data under investigation [19]. The aim is to select (not to lose) those points (outliers) which would be excluded if the standard sampling methods were used.

Thus, in this research, we propose a new approach for data subset selection and for massive data visualization. The main idea of the proposed selection is to take into account the density of points. This is implemented via data clustering, i.e. the sum of distances from each data point to the centre per cluster and the number of points per cluster are estimated.

The proposed data subset selection can be summarized as follows:

Step 1: data clustering is performed to divide the high-dimensional points of the $n \times m$ data matrix X into k clusters;

Step 2: for each cluster, the sum of distances ($sumD_i$) from the cluster centre to each point is calculated; the number of points N_i, $i = 1, \dots, k$ per cluster is calculated;

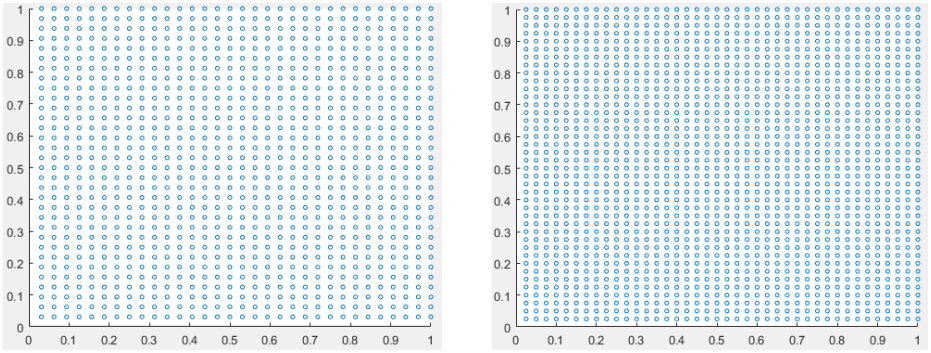
Step 3: the size s of data subset is determined, i.e. the number of points that will be selected as candidates for visualization is defined;

Step 4: the ratio ($r_i = sumD_i/N_i$) is calculated. The number of points to be selected from each cluster into the data subset is calculated by the formula $N'_i = \frac{r_i \times s}{v}$, where $v = \sum_i^k r_i$, and s – the size of data subset;

Step 5: the initial data subset X_S of size $s \times m$ is selected;

Step 6: dimensionality of points of the initial data subset X_S is reduced by a projection technique and the matrix Y_S is obtained. The size of matrix Y_S is $s \times d$.

The size s of the data subset (*Step 3*) is the number of instances (points) that will be candidates for visualization. We recommend selecting $s = 1000$ as the maximal number of points. 1000 points yield a good balance between data representation and quality of the final mapping. This fact is illustrated in Figure 1. Here two-dimensional points are evenly distributed on the 1×1 rectangle. We see that the points are not overlapping. However, when the number of points is equal to 1600 (Figure 1b), they are very close to each other. When visualizing real-world data, the points are not so evenly distributed and they can concentrate in groups, and consequently do not scatter so widely and sparsely. If 1000 points of real-world data are taken for visualization, they will be more concentrated than that in Figure 1a, and it is maybe that a part of them will be overlapping. Such visual overlapping will be eliminated in the next step of the proposed visualization approach (see subsection 2.2).



a) b)

Figure 1. Visualization of points in the interval [0,1]. The size of data subset: a) $s = 1024$, b) $s = 1600$.

There may be cases that the number of points N_i' to be selected from a cluster, calculated in *Step 4*, is larger compared with the actual number of points N_i in a cluster i. In those cases, we suggest to select all the points from the cluster and increase the number of points to be selected from the remaining clusters according to their ratio r_i. The value of increase is $\delta = N_i' - N_i$. Table 1 provides an example where the number of points to be selected from the first and second clusters is increased with respect to their ratio r_i when the *Random* data set is analysed ($n = 2\,515, m = 10, k = 3$, the size of data subset is $s = 1\,000$). It can be seen that according to the ratio $r_3 = 4$ of the third cluster, $N_3' = 679$ points should be selected to the data subset, but the third cluster contains only $N_3 = 15$ points. Thus, we select all the 15 points from the third cluster to the data subset and increase the number of points to be selected from the remaining clusters by $\delta = 664$ points with respect to their ratios r_1 and r_2.

Cluster (i)	Number of points per cluster (N_i)	$sumD_i$	Ratio (r_i)	Number of points from cluster (N_i')	Number of points from cluster (N_i') (increased)
1	500	461	0.922	156	479
2	2 000	1 941	0.971	165	506
3	15	60	4	679	15

The number of points to be selected from 1st and 2nd clusters should be increased by $\delta = 664$ points

Table 1. Example of increasing the number of points to be selected from clusters

Projection of the initial data subset X_S in *Step 6* can be found by various dimensionality reduction techniques. In this paper, well known projection technique – multidimensional scaling is used to reduce the dimensionality of the initial data subset [20]. In this paper, the dimensionality of a data subset is reduced to two ($d = 2$) due to the visualization purpose.

1.2. Data visualization without overlapping

As usual, points are overlapping in a scatter plot, when their huge amount is visualized. If it is necessary to identify each point (or position of the point) we need to eliminate the points which visually overlap and cover each other. Let the input be a subset of points of reduced dimensionality Y_S and the output be a subset to be visualized $Y_L, s \leq l$. The proposed data subset visualization without overlapping can be summarized as follows:

Step 1: the values of features of the initial subset of reduced dimensionality Y_S should be normalized in the range [0, 1], so that the minimal value of each feature were equal to 0, and the maximal one were equal to 1. The normalization is performed in order to set the same value of the *threshold* (see *Step 2*) for all data sets and to be able to compare the results obtained;

Step 2: the normalized data subset points of the reduced dimensionality Y_S, are re-selected with a certain *threshold* t. The *threshold* controls the density of points. The re-selection is performed in the following way: the distance matrix Δ for the points of reduced dimensionality is calculated; if the distance from one point to another is less than t, then the point is eliminated from the initial normalized data subset Y_S,. The size of the final data subset Y_L is $l \times m$ ($l \leq s, d < m$);

Step 3: the data subset Y_L is visualized in a scatter plot.

3. EXPERIMENTAL RESULTS

To show the performance of the proposed visualization approach, some experimental investigations are carried out. 10 benchmark test data sets are visualized by the proposed approach. *Magic gamma telescope*, *Waveform*, *Wine quality*, *Letter recognition*, *Musk*, *Animals*, *Skin segmentation*, *Image segmentation*, *Yeast* data sets are taken from UCI Machine Learning Repository [21], a *Random*

data set is generated by us, where the numbers are uniformly distributed in the intervals [0, 1.0] – 1st cluster; [1.5, 2.5] – 2nd cluster; [6.0, 11.0] – 3rd cluster. Each data set analysed has some specific characteristics. Short descriptions of the data sets are presented in Table 2.

Multidimensional data sets have been clustered by a *k-medoids* method [15]. The number of clusters *k* has been determined by the Calinski-Harabasz clustering evaluation criterion [22]. To evaluate the *proposed approach* of the data subset selection, we compare it to the *stratified* sampling. In the stratified random sampling, a population is divided into smaller sub-groups called strata. The random sampling is applied within each subgroup or stratum [23]. In this paper, the relevant strata are identified using the *k-medoids* method.

Name	n	m	k
Random	2 515; 15 020	10	3
Magic gamma telescope	18 905	10	3
Waveform	5 000	21	3
Wine quality	3 961	11	2
Letter recognition	18 668	16	2
Musk	6 581	166	3
Animals	16 384	72	4
Skin segmentation	51 444	3	2
Image segmentation	2 086	19	4
Yeast	1 453	8	2

Table 2. Data sets (n – number of instances, m – number of features, k – number of clusters).

Table 3 shows the comparison of two subset selection methods (*stratified* sampling and the *proposed approach* of subset selection). Considering paper size limit only three data sets (*Random, Magic gamma telescope, Musk*) are analysed deeply. The columns "*Subset by the proposed approach*" and "*Stratified subset*" provide the information how many points (N_i') should be selected from each cluster to a data subset. The comparison shows that it is important how the subset is obtained, i.e. the numbers of points in clusters of subsets differ especially when *Random* and *Magic gamma telescope* data sets are analysed.

The visualization results of the *Random* data set and its subsets, selected by different methods, i.e. the *stratified* sampling and the *proposed approach* of subset selection, are presented in Figure 2. The images show that the distribution of the points of the *Random* data subset, obtained by the *proposed approach*, is similar to that of the full data set, while the *stratified* subset does not retain the structure of the third cluster (the red points), i.e. only one point from the third cluster is selected to the data subset. 526 points of the *Magic gamma telescope* data set are selected from the second cluster (the green points) by the *proposed approach* and 194 points by the *stratified* method. The opposite situation is with the third cluster (the red points) 195 and 424 points are selected, respectively. Our *proposed approach* takes into account the density of points, the higher the ratio r_i, the more points from the cluster are selected to the data subset and vice–versa. It is obvious from Table 3 and Figure 2 that the structure of the *Musk* data subsets, obtained by the compared methods, differs insignificantly ([323; 335; 342] and [383; 354; 263]). That is why the ratio and the number of points per cluster is similar for all clusters.

The next stage of massive data visualization is elimination of overlapping points. In this paper, we have determined the *threshold* values t according to which overlapping of points can be eliminated. Figure 3 shows the comparison of two *threshold* values with ten different data sets. The size of all data subsets is $s = 1000$, and all the data subsets have been normalized. Figure 3 illustrates the number of points in the subset that finally will be visualized without overlapping with various values of the threshold. It is obvious that a smaller value allows us to obtain a smaller number of points. However, this fact depends on the particular data set. Figure 2 shows some examples of visualization of the subset points without overlapping. We recommend a *threshold* value which varies in the interval $[0.0075, 0.01]$, furthermore, the re-selection should be applied to normalized data subsets. The results have shown that the point overlapping is eliminated with $t = 0.01$ for all data sets.

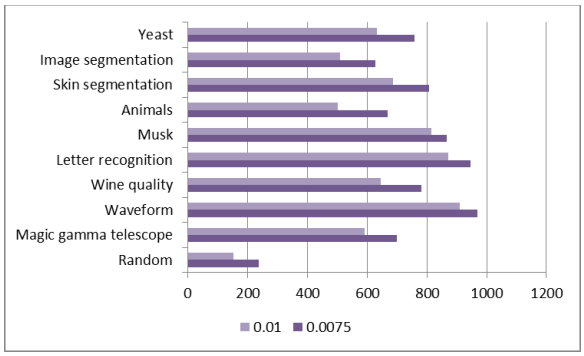


Figure 3. Comparison of two *threshold* values ($t = 0.01, t = 0.0075$) .

It should be noted that, if we are interested in a particular data set point that has not been selected as a member of the data subset and not visualized in the scatter plot, we can find its nearest neighbour in a high dimensional space and its projection in the 2D space. The position of projection of the nearest neighbour approximately shows where a certain point should be.

Data set	Cluster (i)	Number of points per cluster (N_i)	$sumD_i$	Ratio (r_i)	Subset by the *proposed approach*	Subset by the *proposed approach* (re-distributed)	*Stratified* subset
Random	1	14 000	13 410	0.96	125	468	932
	2	1 000	1 048	1.05	136	512	67
	3	20	114	5.7	739	20	1
Magic gamma	1	7 218	533 352	13 969	279	X	382
	2	3 672	511 559	26 342	526	X	194
	3	8 015	413 549	9 754	195	X	424
Musk	1	2 518	2 259 955	898	323	X	383
	2	2 332	2 172 782	932	335	X	354
	3	1 731	1 644 631	950	342	X	263

X – re-distribution of the number of points to be selected is not applied

Table 3. Selection of a data subset by two different methods.

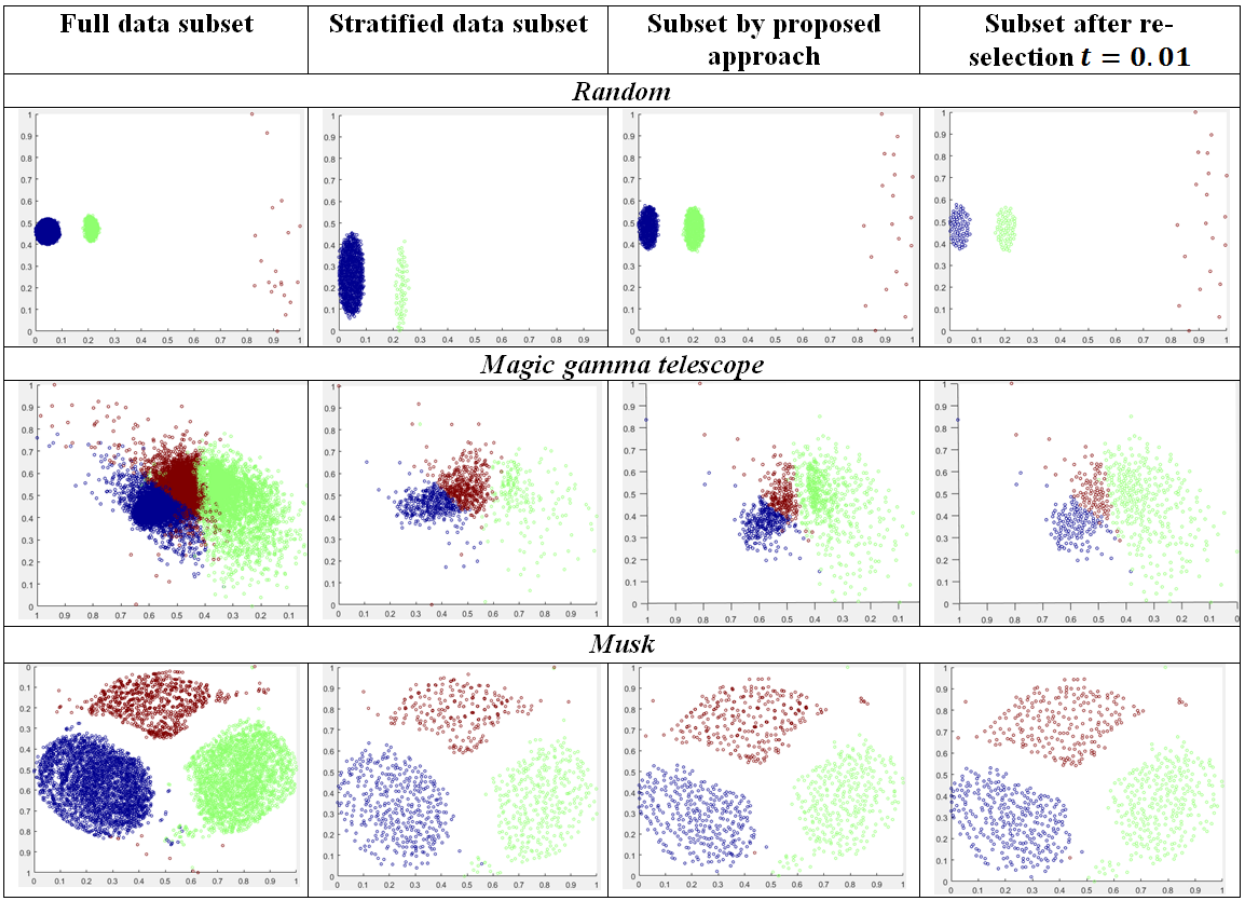


Figure 2. Comparison of scatterplot using various selections of a data subset.

4. CONCLUSIONS

In this paper, we have proposed a new strategy of massive data visualization. The *proposed approach* consists of two stages – data subset selection and visualization of the data subset without visual overlapping. The proposed data subset selection is efficient in terms of preserving outliers in a data subset. We have shown that the precisely selected subset can represent the massive data set well. The investigation results have shown that overlapping of subset points can be eliminated by applying a re-selection of points with a certain *threshold*. We have proposed two *threshold* values. Using them, the algorithm eliminates the overlapping of points. The position of a certain point that is not a member of a subset can be found by the nearest neighbour in a scatter plot. We conclude that the *proposed approach* can be applied as a new way of visualizing massive data sets.

5. REFERENCES

[1] L. P. J. van der Maaten, E. O. Postma and H. J. van den Herik, "Dimensionality reduction: a comparative review," 2009.

[2] D. Cook, E. K. Lee and M. Majumder, "Data Visualization and Statistical Graphics in Big Data Analysis," *Annual Review of Statistics and Its Application,* vol. 3, pp. 133-159, 2016.

[3] J. Bernatavičienė, D. Dzemyda, O. Kurasova, V. Marcinkevičius, V. Medvedev and P. Treigys, "Cloud Computing Approach for Intelligent Visualization of Multidimensional Data," *Advances in Stochastic and Deterministic Global Optimization. Optimization and Its Applications,* vol. 107, pp. 73-85, 2016.

[4] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani and S. U. Khan, "The rise of "big data" on cluod computing: Review and open research issues," *Information systems,* vol. 47, pp. 98-115, 2015.

[5] D. Laney, "3D data management: controlling data volume, velocity and variety," *Meta group,* 2001.

[6] S. Sagiroglu and D. Sinanc, "Big data: A review," in *2013 International Conference on Collaboration Technologies and Systems (CTS)*, San Diego, 2013.

[7] R. Agrawal, A. Kadadi, X. Dai and F. Andres, "Challenges and Opportunities with Big Data Visualization," in *7th International Conference on Management of computational and collective intElligence in Digital EcoSystems*, Caraguatatuba, 2015.

[8] P. Pawliczek and W. Dzwinel, "Interactive Data Mining by Using Multidimensional Scaling," *Procedia Computer science,* vol. 18, pp. 40-49, 2013.

[9] K. Paulauskienė and O. Kurasova, "Analysis of dimensionality reduction methods for various volume data," in *Information Technology. 19th Interuniversity Coneference on Information Society and University Studies (IVUS 2014)*, Kaunas, 2014.

[10] K. Paulauskienė and O. Kurasova, "Projection error evaluation for large data sets," *Nonlinear Analysis: Modeling and Control,* vol. 21, no. 1, pp. 92-102, 2016.

[11] M. Gounder, V. Iyer and A. Mazyad, "A survey on business intelligence tools for university dashboard development," in *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, Sultanate of Oman, 2016.

[12] A. Shukla and S. Dhir, "Tools for Data Visualization in Business Intelligence: Case Study Using the Tool Qlikview," *Information Systems Design and Intelligent Applications,* vol. 434, pp. 319-326, 2016.

[13] J. Miller, Big Data Visualization, Birmingham: Packt Publishing, 2017.

[14] S. G. Archambault, J. Helouvry, B. Strohl and G. Williams, "Data visualization as a communication tool," *Library Hi Tech News,* vol. 32, no. 2, pp. 1-9, 2015.

[15] J. Han and M. Kamber, Data Mining Concepts and Techniques, San Francisco: Morgan Kaufman Publishers, 2006.

[16] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters,* vol. 31, no. 8, pp. 651-666, 2010.

[17] S. L. Lohr, Sampling: Design and Analysis, 2nd edition, Boston: Brooks/Cole, 2010.

[18] G. S. Maddala, Introduction to Econometrics 2nd ed., New York: MacMilan, 1992, p. 89.

[19] O. Maimon and L. Rockah, Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers, Kulwer Academic Publishers, 2005.

[20] I. Borg and P. Groenen, Modern Multidimensional Scaling: Theory and Applications, 2 ed., New York: Springer, 2005.

[21] M. Lichman, "UCI Machine Learning Repository," University of California, Irvine, School of Information and Computer Sciences, 2013. [Online]. Available: http://archive.ics.uci.edu/ml. [Accessed 1 1 2017].

[22] U. Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 24, no. 12, pp. 1650 - 1654, 2002.

[23] Y. Ye, Q. Wu, J. Z. Huang and L. X. Li, "Stratified sampling for feature subspace selection in random forests for high dimensional data," *Pattern Recognition,* vol. 46, no. 3, pp. 769-787, 2013.

Evaluation of an object detection system in the submarine environment

Farah Rekik
Computer and embedded system laboratory, National Engineering School of Sfax
3052 Sfax,Tunisia
farah.rekik@enis.tn

Walid Ayedi
Computer and embedded system laboratory, National Engineering School of Sfax
3052 Sfax,Tunisia
ayadiwalid@yahoo.fr

Mohamed Jallouli
Computer and embedded system laboratory, National Engineering School of Sfax
3052 Sfax,Tunisia
mohjallouli@gmail.com

ABSTRACT

The object detection in underwater environment requires a perfect description of the image with appropriate features, in order to extract the right object of interest. In this paper we adopt a novel underwater object detection algorithm based on multi-scale covariance descriptor (MSCOV) for the image description and feature extraction, and support vector machine classifier (SVM) for the data classification. This approach is evaluated in pipe detection application using MARIS dataset. The result of this algorithm outperforms existing detection system using the same dataset. Computer vision in underwater environment suffers from absorption and scattering of light in water. Despite the work carried out so far, image preprocessing is the only solution to cope with this problem. This step creates a waste of time and requires hardware and software resources. But the proposed method does not require pretreatment so it accelerate the process.

Keywords

Object detection; pipe detection; underwater imaging; descriptor; classifier.

1. INTRODUCTION

The investigation of the underwater environment is a concern in many sectors such as oceanographic research[Eri01a], military applications [Baz00a] and recently the offshore wind energy[Vol17a] with the desire to exploit natural resources for more than 1000 meters deep. Automatic detection of manmade object laying on the seafloor is an important project for international marine research. A great attention is paid to this area. Autonomous Underwater Vehicles (AUV), used for this kind of project, are equipped by sonar system.

Compared to sonar, vision is not widely used in underwater research. This is due to degradation of image quality caused by absorption and scattering of light in water. But sonar suffers from several problems like cost resolution and complexity of use. Therefore there is a need for additional investigation to assess the actual potential of visual perception in underwater environments. Actually, the underwater video is increasingly used as a complementary sensor to the sonar especially for detection of objects or animals. However, the underwater images present some particular difficulties including natural and artificial illumination, color alteration and light attenuation [Bat10a]. Therefore in the detection of underwater objects it is impossible to take only color as a detection criterion, but location, shape and color information must be combined.

Object detection is based on the extraction of discriminative features. This extraction is done by the meaning of a descriptor which describes the image through a characteristic vector using specific features which differ from one descriptor to another.

In this paper the detection algorithm that we will adopt, take into account structural and content features such as pixel coordinates, intensity, gradients, etc. Feature extraction is the first step in which MSCOV [Aye12a] descriptor combines multi-scale features into a covariance matrix. Then, the classification of those matrixes is done by the SVM classifier to generate the detection model. The experiments are performed on MARIS [Ole15a] datasets and compared with object detection algorithm based on multi-scale graph-based segmentation (MGS) [Kal15a] and pixel-feature clustering (PFC) [Kal14a] method.

The rest of the paper is organized as follows. Section 2 reviews recent solution. In section 3, we describe the method proposed for better underwater object detection. The experimental setup and experimental results are presented in section 4. The paper concludes in section 5.

2. RECENT SOLUTIONS

Computer vision in underwater environment

In recent years, the interest of the scientific community in underwater computer vision has increased, taking advantage of the evolution of sensor technology and image processing algorithms'.

In [Weh14a], Wehkamp and Fischer described a workflow for stereoscopic measurements for marine biologists by providing instructions on how to assemble an underwater stereo-photographic system with two digital consumer cameras with underwater calibration. However their study didn't take into account the degradation of underwater images, and absorption and scattering phenomena in water. Artificial vision applications in underwater environments include detection and tracking of submerged artifacts. In [San13a] an embedded stereo-vision system for underwater object detection was presented based on FPGA technology. The system achieved a throughput of 26.56 frames per second (800x480 pixels). It is a high performance system in the hardware and a software levels. A standard in-air calibration procedure was adopted. Ortiz et al. proposed in [Ort02a] a single-camera vision system for real-time underwater cable tracking to detect power cables laid on the seabed.

Underwater object recognition using computer vision is difficult due to the lighting condition of such environment. Kim et al. [Kim12a] present a vision-based object detection method based on template matching and tracking for underwater robots using artificial objects. Results are not performed in submarine context, but only in a swimming pool. An important difficulty in the processing of underwater images came from the problem of attenuation of light in water. Bazeille et al. [Baz00a] cope with this problem and discuss the color modification in underwater environments and experimentally assess the performance of object detection based on color.
According to the pipe detection in underwater environment we will detail two approaches with which we will compare our work: Pixel-Feature Clustering (PFC) and Multi-scale Graph-based Segmentation (MGS). PFC algorithm performs clustering on the pixels of the image according to local pixel features and next selects the connected components according to the shape [Kal14a]. Features used by PFC, which are extracted from each pixel, consists of the color channels of HSV space, respectively hue , saturation and value , and of the gradient response to a Sobel filter. MGS algorithm belongs to the popular graph-cut approach representing the image as a grid graph. It first exploits color uniformity to enable better partitioning of the image into homogeneous regions, and then finds the shape searching for regular contours [Kal15a]. PFC and MGS algorithms reach a good detection rate using a small dataset. However they must their performance must be proved in a larger benchmark.

Image descriptors

2.1.1 Global descriptor

Global description consists on describing the whole image by their characteristics taken from each pixel. The color histogram is the best-known descriptor in this context. It represents the distribution of intensities or color components of the image. The most used global descriptors are statistics descriptors. They are determined following a frequency filtering, starting from the co-occurrence matrices, or from the first-order or high-order statistics.

Global descriptors are known for their speed and simplicity of implementation. The combination of several global characteristics can achieve good results. However, global description suffers from several problems. It implicitly assumes that the entire image is related to the object. Thus, any incoherent object would introduce noise affecting the description of the object. This limitation encourages the use of local descriptors, or even regions.

2.1.2 Local descriptor

Local description is based on the identification of local points of interest with a vector of attributes, and on the use of local descriptors which characterize only a small part of the image. SIFT[Par06a] is the most popular local descriptor.

The most interesting property of this descriptor is its robustness to the image transformation. The problem is that the objects are represented by a variable number of points of interest, whereas the classifiers require a vector with a fixed size as input. As for the descriptors by region, the feature vector is fixed, which is more suitable for classifiers.

2.1.3 Region descriptor

This approach consists in decomposing the image into a set of fixed or variable size regions and then characterizing each of these regions. The decomposition is done in a predictable way in order to make the regions' characteristics homogeneous with each other. These descriptors have recently been successful in several applications. Covariance descriptor [Tuz06a] is mainly used in human detection and re-identification. On the other hand, this descriptor has some limitations. Indeed, it implicitly assumes that the whole region is connected to the object to be modeled while the latter may have an incoherent shape.

MSCOV came to improve this descriptor by the adjustment of the trade-off between the local and the

global description of the objects. This descriptor will be detailed in the next section.

3. PROPOSED SOLUTION

Object detection approach

Detecting object in underwater environment requires much attention. The purpose is identifying the region of interest which contains the target. Objects are characterized by the variation of textures, colors and shape. Those features will be adopted to create a model. The descriptor makes it possible to transform an image into a characteristics vector by extracting discriminative information of the class sought, in order to train the classifier.

The classifier is created from the database of positive and negative samples to define a detection model. Positive samples are images containing object and Negative samples are those which do not contain an object.
Then the image of the scene is scanned to find the candidates that will be compared to the database through the classifier.

The structure of an object detector is defined by the following figure.

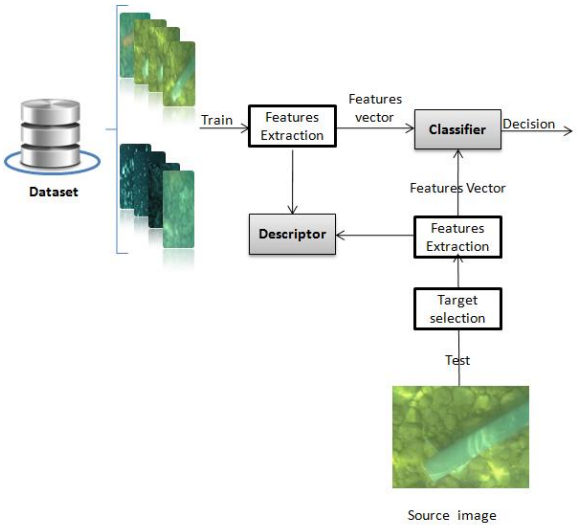


Figure 1. Detector approach

Multi-scale covariance descriptor

The descriptor adopted for this work is the multi-scale covariance descriptor [Aye12a]. It is based on the quadtree structure which explains the multi-scale aspect.

This structure is widely used for image representation in computer vision applications [Kim00a], [Lin01a], [Mal99a]. It is also used to store and index image characteristics and region of interest.

The quadtree represents a hierarchical structure constructed by recursive divisions of the image in four disjoint quadrants with the same size, according to homogeneity criterion, until a stop condition is reached. Fig 2.a presents a quadtree applied to a frame of the dataset used in this work.

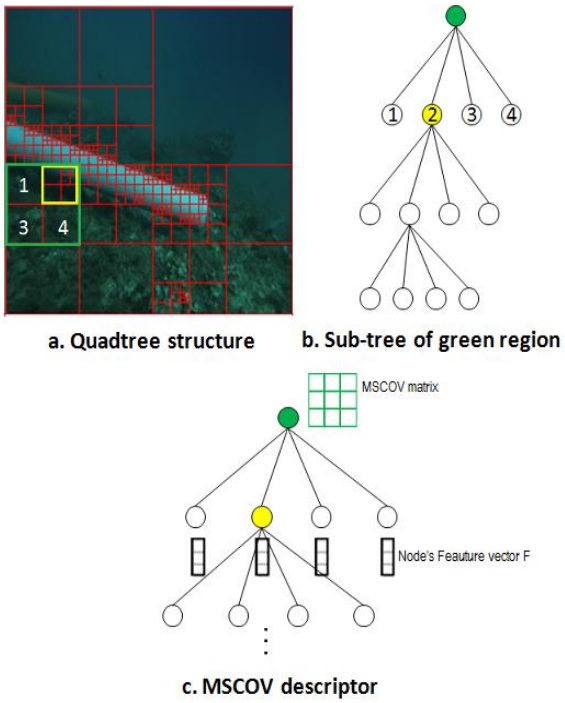


Figure 2. MSCOV and quadtree representation

Each image quadrant is represented by a quadtree node and the root node represents the whole image. Fig2.b represents a sub-tree of the green region in the pipe image.

As described in Fig2.c the MSCOV descriptor characterizes a quadrant image through the characteristics stored in its associated node. Each node stores a features vector defined by:

$$F = \left[x, y, Y, I, C_r, C_b, I_x, I_y, grad, mag \right]$$

Where x is the x location of the corresponding image quadrant, y is the y location of the corresponding image quadrant, Y is the node level, I is the I grayscale intensity value (the luminance component), C_r is the red chrominance component, C_b blue chrominance component, I_x is the norm of the first order derivatives in x, I_y the norm of the first order derivatives in y, grad is the gradient and mag is the magnitude.

Features are arranged in two groups. The structural characteristics which are related to image data location (x, y and Y), and the content characteristics that are derived from the color information (I, Cr, Cb, mag, grad).

Feature vectors in each node are combined into a covariance matrix defined by:

$$C_r = \frac{1}{N_r - 1}\sum_{c=1}^{N_r}(F_c - m)(F_c - m)^T$$

Where Nr is the number of the node in the sub-tree of r, m the mean of the nodes features and Fc the feature vector of the node c descendant of r. This structure is nominated as ‘‘Image Quadtree Features’’ (IQF).

The multi-scale covariance descriptor provides two main advantages. In fact, the decomposition into a quadtree makes it possible to capture the region of interest of the image (points of interest), and consequently it reduce the impact of noise and background information on the description of the object. Therefore the pre-treatment step is canceled. In addition, quadtree is used as a multi-level structure to extracts image features from different scales. It thus makes it possible to optimize the compromise between the local and the global description of the object.

Fig 3 and 4 show the difference between the ordinary description based on pixels of the whole region, and the multi-scale description based on the nodes. The latter is based on both the global information and the local information of the object, and focuses implicitly on the object described.

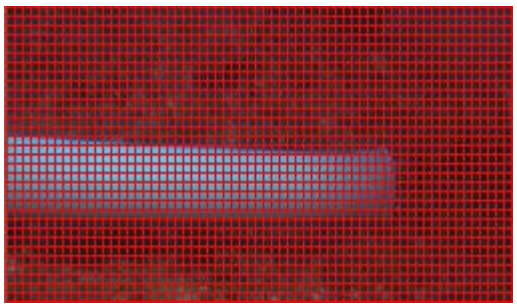


Figure 3. Global description from pixel

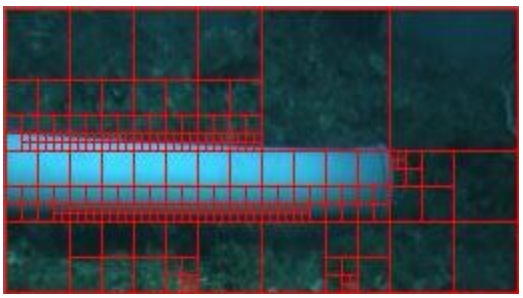


Figure 4. Local description from nodes

4. EXPERIMENTAL RESULT

This section is organized as follows. First we present the adopted evaluation metric. Then we describe the dataset used in the experimentation. Finally we compare our results to Fabjan et al[Kal15a] works using the same dataset.

Evaluation metric

Precision and recall and F-measure are the appropriate metric to evaluate the detection accuracy.

They have always been used for the evaluation of pattern detection algorithms. They are defined as follows:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Where TP is the true positive, FN is the false negative and FP is the false positive. TP is the number of images real positive and predicted positive. FP is the number of image real positive and predicted negative. FN is the number of image real negative and predicted positive.

F-Measure is also a measure of a test's accuracy. It is the harmonic-mean of precision and recall. It’s defined by:

$$F_{Measure} = \frac{2.Reall.Precision}{Recall + Precision}$$

Maris dataset

Maris dataset [Ole15a] is used to evaluate the performance of the proposed approach.

This dataset is acquired using a stereo vision system near Portofino (Italy). It provides images of cylindrical pipes with different color submerged at 10 m deep. The dataset include 9600 stereo images in Bayer encoded format with 1292x964 resolution, and it include positive (frames containing a pipe) and negative (frames presenting only the background) frame Fig 5.

Figure 5. Maris dataset simples’

From this dataset we will use 305 frames, presented in Table 1, to train the classifier.

Graphics	Samples number	Positive simples	Negative simples
Training data	305	205	100

Table 1. Training dataset

This set of data presented previously contains exactly the same frames used by Fabjan et al in [Kal15a] in order to make subjective comparison. For our test we divide those frames in two classes. The first class contains positive frames, and the second one contains negative frames.

Results

Our work is compared to PFC and MGS algorithms using the same samples of Maris dataset.

This dataset is taken using two stereo camera a left and a right camera providing pairs of frame. We use only the left camera frames after been resized to350x250 pixels. Frames contain one or more pipe with different color. PFC and MGS take only one pipe in the foregrounds. But the proposed solution can take more than one. Therefore the performance will be grater.

Table 2 illustrates the performance parameters for the tree methods.

	MSCOV	PFC	MGS
TP	205	179	177
TN	96	91	94
FP	**4**	11	7
FN	**0**	24	27
Precision	98,08%	94.2%	96.2%
Recall	100%	88.2%	86.8%
Detection Accuracy	**98,68%**	89.2%	88.9%
1-FPRate	96%	89.2%	93.1%
F-Measure	99,03%	91.2%	91.2%

Table 2. Detection result for PFC, MGS and MSCOV algorithms on Maris dataset

The MSCOV detection accuracy is the best. There is an improvement of 10% in the detection accuracy. The Recall metric reach 100% for MSCOV, but never exceed 89% for the others. As a result the proposed approach outperforms PFC and MGS algorithms.

5. Conclusion

In this paper we have presented a novel underwater object detection algorithm based on multi-scale covariance descriptor for features extraction, and SVM classifier for data classification. This algorithm has been tested with Maris dataset and compared to PFC and MGS algorithms using the same frames. The experimental results show that it outperforms compared methods, and detection accuracy can reach 98%. In future work we will use a larger dataset in order to generalize the result and show that this adopted approach is valid in the submarine environment.

We will also work on descriptor parameters to improve the detection accuracy. There are two principal parameters: the homogeneity error tolerance ε and α the precision degree of description.

6. ACKNOWLEDGMENTS

Our thanks to RIZZINI Dario Lodi for allowing us to use Maris dataset and to compare our approach with theirs.

7. REFERENCES

[Aye12a] Ayedi, W., Snoussi, H., & Abid, M. A fast multi-scale covariance descriptor for object re-identification. Pattern Recognition Letters, 33(14), pp.1902-1907, 2012.

[Bat10a] Barat, C., & Phlypo, R. A fully automated method to detect and segment a manufactured object in an underwater color image. EURASIP Journal on Advances in Signal Processing, 568092, 2010.

[Baz00a] Bazeille, S., Quidu, I., & Jaulin, L. Identification of underwater man-made object using a colour criterion. In Conference on Detection and Classification of Underwater Targets p. xx,2007.

[Eri01a] Eriksen, C. C., Osse, T. J., Light, R. D., Wen, T., Lehman, T. W., Sabin, P. L., ... & Chiodi, A. M. (2001). Seaglider: A long-range autonomous underwater vehicle for oceanographic research. IEEE Journal of oceanic Engineering, 26(4), 424-436.

[Kal14a] Kallasi, F., Oleari, F., Bottioni, M., Rizzini, D. L., & Caselli, S. Object detection and pose estimation algorithms for underwater manipulation. In Advances in Marine Robotics Applications (AMRA)-in International Conference on Autonomous Intelligent Systems (IAS) pp. 1-7, 2014.

[Kal15a] Kallasi, F., Rizzini, D. L., Oleari, F., & Aleotti, J. Computer vision in underwater environments: A multi-scale graph segmentation approach. In OCEANS 2015-Genova pp. 1-6, 2015.

[Kim12a] Kim, D., Lee, D., Myung, H., & Choi, H. T. Object detection and tracking for autonomous underwater robots using weighted template matching. In OCEANS, pp. 1-5, 2012.

[Kim00b]Kim, H. K., & Kim, J. D. Region-based shape descriptor invariant to rotation, scale and translation. Signal Processing: Image Communication, 16(1), p. 87-93,2000.

[Lin01a] Lin, S., Ozsu, M. T., Oria, V., & Ng, R. An extendible hash for multi-precision similarity

querying of image databases. In VLDB pp. 221–230,2001.

[Mal99a] Malki, J., Boujemaa, N., Nastar, C., & Winter, A. Region queries without segmentation for image retrieval by content. In International Conference on Advances in Visual Information Systems pp. 115-122. Springer Berlin Heidelberg, 1999.

[Ole15a] Oleari, F., Kallasi, F., Rizzini, D. L., Aleotti, J., & Caselli, S. An underwater stereo vision system: from design to deployment and dataset acquisition. In OCEANS 2015-Genova pp.1-6,2015.

[Ort02a] Ortiz, A., Simó, M., & Oliver, G. A vision system for an underwater cable tracker. Machine vision and applications, 13(3), pp.129-140 2002.

[Par06a] Park, U., Jain, A. K., Kitahara, I., Kogure, K., & Hagita, NVise: Visual search engine using multiple networked cameras. In *Pattern Recognition,. 18th International Conference on* IEEE Vol. 3, pp. 1204-1207,IEEE,2006.

[San13a] Sánchez-Ferreira, C., Mori, J. Y., Llanos, C. H., & Fortaleza, E. Development of a stereo vision measurement architecture for an underwater robot. In Circuits and Systems (LASCAS),IEEE Fourth Latin American Symposium on pp. 1-4, 2013.

[Tuz06a] Tuzel, O., Porikli, F., & Meer, P. Region covariance: A fast descriptor for detection and classification. In *European conference on computer vision* Springer Berlin Heidelberg, pp. 589-600, 2006.

[Vol17a] Voltaire, L., Loureiro, M. L., Knudsen, C., & Nunes, P. A. (2017). The impact of offshore wind farms on beach recreation demand: Policy intake from an economic study on the Catalan coast. Marine Policy, 81, 116-123.

[Weh14a] Wehkamp, M., & Fischer, P. (2014). A practical guide to the use of consumer-level digital still cameras for precise stereogrammetric in situ assessments in aquatic environments. Underwater technology, 32(2), pp.111-128, 2014.

Ellipse Detection in Very Noisy Environment

Pierre Zoppitelli
Pierre.Zoppitelli@univ-amu.fr

Sébastien Mavromatis
Sebastien.Mavromatis@univ-amu.fr

Jean Sequeira
Jean.Sequeira@univ-amu.fr

Aix-Marseille University
Polytech Sud - LSIS - Case 925
163, avenue de Luminy
13009 Marseille, France

ABSTRACT

Ellipse detection is a major issue of image analysis because circles are transformed into ellipses by projective transformations, and most of 3D scenes contain circles that are significant for understanding them (mechanical parts, man-made objects, interior decoration,). Several algorithms have been designed and published, some of them very recently, to characterize ellipses within images and they are very efficient in most classical situations. But they all fail in some specific cases that regularly happen as, for example, when the noise in the image is such as it produces dashed ellipses, or when they are very flat, or when only small parts of them are visible. We propose an algorithm that brings a solution in such cases, even if it is not more efficient than the other ones in classical situations. Hence, it can be used as a complement of other algorithms when we want to detect ellipses in a robust way, i.e. in all situations. This algorithm takes advantage of a property of ellipses related to their tangent lines, without any assumption on edge connectivity: primitives are designed to characterize the possibility for a point and an orientation to locally represent an edge; these primitives are not connected and their global analysis enables to obtain the center location and the three other parameters of ellipses that can be drawn through this set of primitives, i.e. that go through some of these points and that have the corresponding tangent lines. A set of tests has been used to measure its robustness.

Keywords

Ellipse detection, Hough Transform.

1 INTRODUCTION

Image understanding requires to extract relevant elements from images and to make them match with the corresponding potential items of a scene model; we also could say that it consists in extracting information from images and associating it with a knowledge representation. Knowledge can be related to color, texture, shape and many other factors, in relation to the problem we want to study. When considering shapes, we can notice that circles are often part of knowledge representation. For example, many road signs are circular, so that a lot of man-made devices that can be seen either inside (boxes, lampshades, clocks,) or outside (traffic circles, pipes,).

Circles are transformed into ellipses by homographies (projective transformations) and it is especially the case of images produces by video cameras. Hence, several ellipses are present in such images and characterizing them is crucial in the frame of robotics, video watching and UAV (Unmanned Aerial Vehicles) applications, for example.

2 STATE OF THE ART

Ellipse properties have been widely studied since the Greek Antiquity, with a lot of interesting results in the seventeenth to nineteenth centuries, especially when considering them in frame of the projective geometry using complex numbers. A lot of papers have been published during the last thirty years on ellipse detection in images, but this topic is particularly crucial and thus, several important papers have been published very recently on ellipse characterization. Let us describe the most recent and significant papers about ellipse detection. Three types of approaches enable to characterize an ellipse within an image, by making an ellipse evolve in the image under constraints, by using a Hough Transform [Dud72], or by linking edges or parts of ellipses.

Using the Hough Transform requires to describe and then to manage a homogeneous 5-dimension space (because an ellipse is defined by 5 parameters), and this is a very complex problem. An alternative solution, based

on this idea, has been proposed in 2005 by Basca et al [Bas05]. Using an ellipse that evolves under constraints is a very classical idea, which is similar to an active contour approach: in 2013, Prasad et al published an improved version of this approach [Pra13]. Finally, some works as those described in Libuda et al [Lib07] propose a method that connects small edges until it produces an ellipse.

Fornaciari et al give an evaluation of these approaches and similar ones in [For14]. In this paper, they propose a new algorithm that improves the "connecting edges" approach by using a Hough Space in which they proceed to a parameter separation (in order to avoid a 5-dimensional space).

Other very recent works have been published (it shows that it is really important to find an efficient solution to this problem), and we mention the algorithm proposed by Lu [Lu15] and that uses a polygonal curve and a likelihood ratio test to stabilize the ellipse evolution.

At the end of 2016, Jia et al propose an original approach [Jia16], based on the "connecting edges" approach but with a major improvement that uses a very powerful property of projective space. In this paper, they compare the performance of this new algorithm with all those algorithms mentioned before and they demonstrate they obtain better results. Thus, we will give a short description of this algorithm in the next lines.

Jia et al use a projective property that is a generalization of the concept of cross-ratio on a polygon (in the case of the cross-ratio, the polygon is reduced to two points) and that enables to provide a Characteristic Number (CN) that is specific of the polygon because it is invariant by projective transformation; in the case of six points located on an ellipse, the CN is +1. In such a way, we can find very easily and very quickly if six points belong to an ellipse or not. They first look for edges (by using a Canny operator [Can86] and a Freeman encoding) and they assemble these edges when they belong to a same ellipse (by using this projective property). Although this algorithm is very efficient compared to the other ones, it fails in some situations that happen frequently. We identified at least, three types of situations in which it is the case: when points on the ellipse are sparsely located and thus do not enable to produce edges (e.g. a road sign partly hidden by branches or the end of pipe partly hidden by a grid), when having a very short arc of ellipse, and when the ellipse is very flat.

We propose an algorithm that takes in charges such situations and that find ellipses, even in very tough conditions. This algorithm is not proven to be better than other ones in classical situations, but it can be used in parallel (as a complement) in the frame of a more robust and reliable detection approach.

3 THE PROPOSED ALGORITHM

Let us show two images illustrating a case in which the algorithms mentioned before often fail.

Figure 1: The road sign is partly hidden by branches

Figure 2: The end of the cylindrical pipe is hidden by a grid

The main reason is that these algorithms require the detection of edges that have a minimum length, and it is not the case in these images. In order to overcome this problem, we decided not to use edges (as sets of connected points) but to base our algorithm on an ellipse local property that is integrated in a global scheme. Let us see in next subsection what is this property.

3.1 A property of ellipse tangent line

Let us consider P_1 and P_2, two points of an ellipse and their corresponding tangent lines T_1 and T_2 in P_1 and P_2; let us also consider I_{12} the middle of $[P_1P_2]$ and J_{12} the intersection point of T_1 and T_2; then, I_{12}, J_{12} and C (center of the ellipse) are on the same line D_{12}.

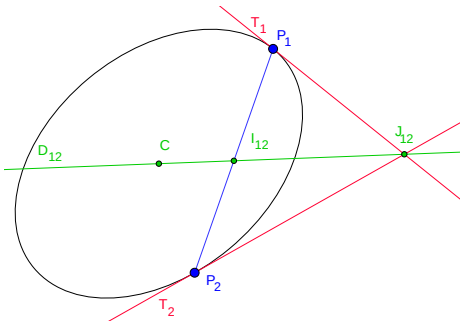


Figure 3: C, I_{12} and J_{12} are on the same line

This property enables to skip from one point to another on the ellipse without any need of point connectivity.

3.2 Primitive definition

We define a primitive that enables to take advantage of this property for characterizing ellipses. Such a primitive consists of a point and a tangent vector. Practically, we prefer to use a representation that is a bit redundant with a point (x and y coordinates) and the normalized coefficients of the tangent line equation (a, b and c, where $a^2+b^2=1$).

Finally: $P=(x,y,a,b,c)$

$(b,-a)$ is a normalized tangent vector, but $(-b,a)$ is another one: the choice of the sign gives us the curvature orientation. The c value is not really necessary but there should be to calculate it at each time we use the tangent line, and thus, it is easier like that.

3.3 The principle of the algorithm

The fundamental idea of our approach is the transposition of a classical reasoning method in mathematics. We first express a necessary condition that drastically reduces the set of potential solutions; then, we look within this set of potential solutions if some of them are effective solutions.

An ellipse can be defined by its center (two parameters) and three other parameters (e.g. the angle of its axis and the lengths of its two main axes). Our method consists in three steps:

1. We look for the approximate location of the center of the potential ellipse (i.e. if an ellipse is "drawn" in the image, its center is necessarily this one)
2. Once a center has been found, we look for the approximate location of a reference ellipse
3. The neighborhood of this reference ellipse is the support for a deeper and strict research of the ellipse, if it exists.

These considerations can be derived with several centers (e.g. several road signs) and for several ellipses for a center (e.g. the inner and the outer ellipses of the outer red band of forbidding road signs).

Within each step, and especially for finding the center of a potential ellipse, we use embedded processes that progressively eliminate wrong solutions and that make the center converging to a quite accurate and reliable location.

3.4 Process homogenization

Several parameters must be adjusted in relation with the size of the ellipses we have to detect, and there may be big ellipses as well as small ones. Adjusting such parameters should be too complex because some of them are correlated. Let's give some parameters that needs to be set : the maximum distance between triplets during the accumulation (if it's too small a big ellipse will not have a distinct peak at its center, if it's too big then it will add a lot of noise and hide small ellipses), the size of the masks during the tangent calculation phase (this mask has to encompass an edge, if it's too big then it will capture too many points outside of the edge, if it's too small then it will be too sensitive to noise), the parameters to compute the spatial connected components (angle between two consecutive edges, distance between primitives, etc...).

Thus, instead of changing their values, we change the size of the ellipses we search, and we keep an optimized set of parameters for a given size of image. Practically, we only look for ellipses that are in a range of 1 to 3, e.g. whose diameter (i.e. the greatest distance between two of its points) is between the $1/2$ and the $1/6$ of the image width.

We first define a strategy for scanning an image with sub-images and we then apply the global process after resizing each of these sub-images to a predefined size. For example, we can apply it to the initial image (to detect "big ellipses"), then to four sub-images (dividing it in each dimension by two), then to nine ones (dividing the initial image in each dimension by three), and so on. Or, it can be driven by a global strategy that only considers some parts of the image that should be relevant.

3.5 Primitive characterization

We can work on color images, on grey-level images and even on binary images (in which edge points are sparsely distributed, and that can be noisy).

In the case we work on a color image, we transform it into a grey level one (e.g. using a linear combination of R, G and B values). Then, we apply a Nagao filter to eliminate a part of the noise (the advantage of this filter is that it replaces "a priori wrong values" by predominant values in their neighborhood). Finally, we apply a High-Pass filter to enhance the contours, and we use a threshold to produce a binary image in which points with value "1" are on the edges.

The next step consists in extracting the "relevant points" from this binary image: these points are those that should to "support" a primitive, i.e. a location at which other points in the neighborhood are roughly along a line (that could be considered as a very local part of a curve).

We define a neighborhood and a set of masks, each of them being associated with an orientation. A mask is a thick segment of a given orientation. For each point of the binary image, we count the number of points covered by each mask and we keep the maximum of these

numbers. If it is greater than a given threshold, we keep the corresponding point as a support for a primitive, else we do not keep it. The purpose of this step is to eliminate points that clearly do not belong to an edge. We obtain a subset of the initial set of points (initial binary image).

For each of these points, we compute the gradient at the corresponding location in the grey level image by applying a Gaussian Blur and a Sobel Operator. In the case we started with the binary image, we obtain an estimation of the tangent vector by using PCA (Principal Component Analysis) on the subset of points covered by the selected mask (it allows a more precise evaluation of the local tangent than by only using the direction of the mask). The drawback of this method is that it's less precise than using the Sobel Operator when the edges are net and precise, however it's more robust to local noise, and thus, might be used even on color image in a very noisy context.

We sort these points by decreasing values of their associated gradient measure; then, we scan this sorted list and we keep only those points whose distance to previous ones (that have already been selected) is greater than a given minimum distance d_{min}. It provides a sparse set of couples "point + gradient" in which two elements are not closer than d_{min}.

Then, we compute a normalized equation of the tangent line at each point from its location and the value of the associated gradient. But, as we mentioned it before, there are two possibilities for this normalized equation, one with (a,b,c) coefficients and the other one with the opposite values $(-a,-b,-c)$ for the coefficients.

We finally choose the set of coefficients that provides a positive result when replacing x and y by the coordinates of the curvature center: we do not know where is the curvature center but we have a reliable information on which side it is, by evaluating of the variance on both sides of the tangent line (the side of the curvature center locally contains pixels of both sides of the curve, and its variance must be greater than on the other side).

3.6 Locating a potential ellipse center

Our goal is not to directly find an ellipse but to focus on specific and structured areas in which the probability of finding an ellipse is different from zero. Such an area could be called an "elliptic band" and is defined by the dilation of a reference ellipse: it looks like a "thick ellipse" but, in some cases (when the reference ellipse is flat), it may look like a "filled ellipse" (even if it is not, mathematically).

Figure 4: An elliptic band (on the right) is defined by the dilatation of a reference ellipse (on the left)

Figure 5: Elliptic band on a flat ellipse

Because this area is structured and does not contain too many pixels, we can then use a dedicated process to efficiently detect an ellipse (or ellipses) if it exists (they exist) in this area.

The first step is to locate the center of this reference ellipse, and we do it by applying the property of tangent lines (mentioned before) to the set of primitives we obtained previously.

Let us assume that an ellipse $\mathcal{E}$ can be associated with a subset $\mathcal{P}_\mathcal{E}$ of $\mathcal{P}$. Then any triplet of primitives (P_i,P_j,P_k) of $\mathcal{P}_\mathcal{E}$ produces a triplet of lines (D_{ij},D_{jk},D_{ki}) that intersect at the center C of $\mathcal{E}$.

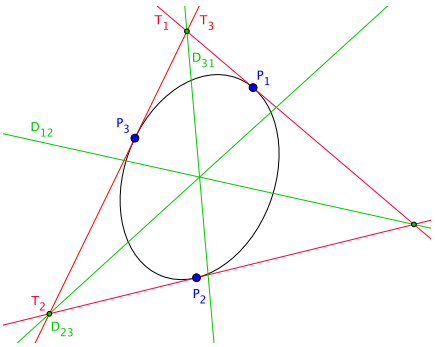


Figure 6: D_{12}, D_{23} and D_{31} intersect in C

Practically, because the primitives cannot be accurately computed, D_{ij}, D_{jk} and D_{ki} do not intersect perfectly in one point and thus, they produce a small triangle $\mathcal{T}r_{ijk}$ that covers the center location.

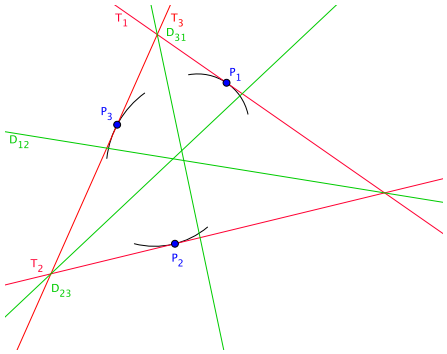


Figure 7: D_{12}, D_{23} and D_{31} do not intersect perfectly

We then use a Hough-like approach in order to characterize the existence of an ellipse and to approximately evaluate its center location. We consider a Hough Space that represents the image at a lower resolution (30x20): we draw triplets of $\mathcal{P}$ (not all the triplets, and under given rules of proximity); when the triangle $\mathcal{T}r_{ijk}$ is small enough (it happens when the three primitives belong to an ellipse $\mathcal{E}$), we evaluate its center in the Hough Space and we increment the corresponding cell. If a relevant maximum appears in the Hough Space, it means that, with a high probability, there is an ellipse $\mathcal{E}$ in the image, we obtain approximately its center C, and we characterize $\mathcal{P}_\mathcal{E}$ by keeping those primitives that "voted" to C.

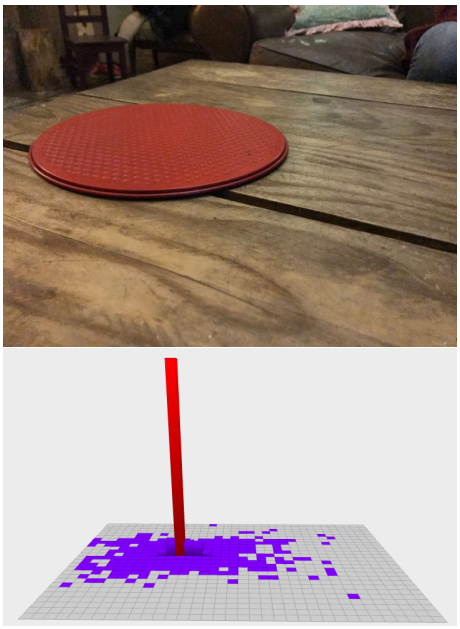


Figure 8: A single ellipse and the Hough Space

We may have to face to complex situations: for example, if there is an elliptic band (and not a single ellipse), then we will have to detect two concentric ellipses (the two boundaries of the band), associated with two subsets of primitives $\mathcal{P}_{\mathcal{E}1}$ and $\mathcal{P}_{\mathcal{E}2}$ that both vote for the same center C. In such a case, the approach described before still works but the resulting center does not appear so clearly, and we had to improve our algorithm in order to be able to provide a correct detection in all cases.

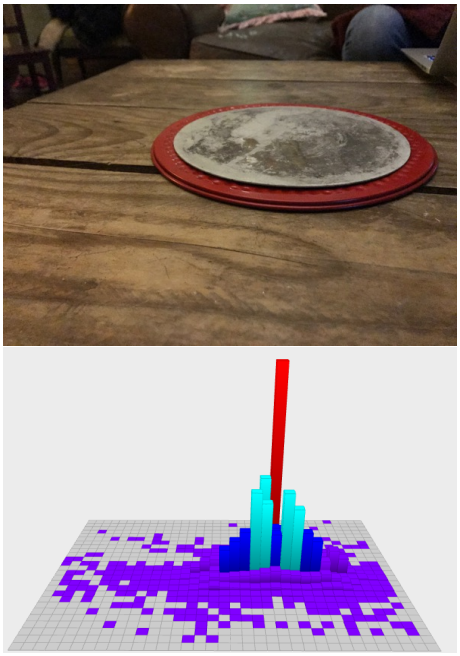


Figure 9: Two ellipses and the Hough center

We have developed an algorithm that enables to separate these two ellipses. At the end of the previous step, we obtain a subset $\mathcal{P}_\mathcal{E}$ and a center C, and we would like to split $\mathcal{P}_\mathcal{E}$ into $\mathcal{P}_{\mathcal{E}1}$ and $\mathcal{P}_{\mathcal{E}2}$. We define a graph – we call it the "Coherence Graph" – whose vertices are the primitives of $\mathcal{P}_\mathcal{E}$ and whose edges joining any couple of primitives have a weight that measures their involvement in finding C (in other terms, the weight of $\mathcal{E}d_{ij}$ is the number of other points P_k that participated with P_i and P_j to the vote for C): the "strong" connected components of this graph produce a link between the primitives of the same ellipses (it is important to notice that the primitives of such a connected component do not necessarily correspond to spatially connected components: it can be a sparse set of points, or two arcs of the same ellipse at opposite sides). We also use a more classical approach that consists in spatially connecting primitives using their proximity and their tangent line coherence.

3.7 Reference ellipse characterization

The Hough Transform provides an ellipse center: we do not directly associate it with the center of the cell that contains the maximum vote, but we use a weighted average with its neighbors. We also obtain a subset of primitives (or more) that participated to the characterization of this center (by means of the "Coherence

Graph" and with the help the spatially connected components). After eliminating outliers and taking care of the relative relevance of the resulting arcs of ellipse, we can characterize a "Reference Ellipse" that is an ellipse globally located close to all the relevant extracted elements.

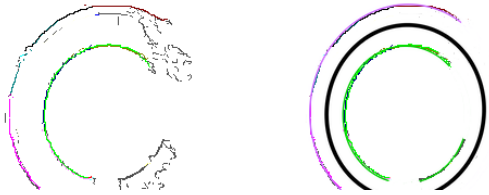


Figure 10: Connected components and reference ellipse

This reference ellipse structures the area in which we will perform a specific and accurate research of an ellipse (or ellipses) located in its neighborhood. This area is generated by the dilation of the reference ellipse, and if there are no topology changes (i.e. if it provides an homotopic transformation), the reference ellipse is its skeleton and is used to drive a dedicated research within this area.

3.8 Validation and configuration analysis

The information extracted all along this process gives us a robust basis to produce an efficient and very fast analysis that reliably and accurately characterizes the situation.

Until now, the algorithm was general and adapted to all cases, independently of the situation (color, texture, single ellipse or elliptic band whose edges are two ellipses,).

Now, we need to introduce knowledge (related to the problem we want to solve) in order to detect what we want to. Typically, we progress along the reference ellipse and we analyze an orthogonal cross section of the corresponding area: the distribution of values along this cross section provides results that can be analyzed globally. It enables to obtain the accurate location of the ellipse or the elliptic band, to characterize the hidden parts, and to refine the analysis inside the ellipse (color,pattern,).

Figure 11: The research area (with dimmed colors) and the extracted arcs of ellipses

4 RESULTS

We did obtain convincing results in some complex situations in which classical algorithms fail, especially for dashed ellipses, for single short arcs of ellipses, and for very flat ellipses.

Let us explain briefly and then illustrate why the algorithm we propose works in these cases when the other ones do not.

About dashed ellipses, the reason is that we do not require any continuity to assemble edges under the constraint of belonging to a same ellipse: we only need a sparse set of primitives that simply indicate a location and an associated orientation.

Most approaches described in the state of the art requires to have parts of ellipses with different orientations, and thus, in most of the cases to have at least more than the half of an ellipse to detect it (in the presentation of the results on different images and the comparison of various algorithms, Jia and al[Jia16] clearly show this problem). Our algorithm does not require this constraint and works even if only a small arc is visible.

In the case of flat ellipses, it is interesting to give some more scientific details. It should seem difficult, and even impossible, to compute the intersection of tangent lines when they are parallel or quite parallel. In fact, it is not the case because all the geometrical elements are considered in the homogeneous space (projective geometry) and thus, there is absolutely no problem to compute the intersection of two parallel lines, and no problem of accuracy when computing the intersection of two lines that are almost parallel.

In the first subsection, we illustrate how the algorithm works in some difficult cases. Then, because we wanted to quantify its limits, we developed a program to test its robustness when evolving toward limit cases, and this is illustrated in the second subsection.

4.1 Some results in critical cases

The results shown in this section were obtained with this set of parameters : 350 pixels for the width of the sub-image, 150 pixels for the maximum distance between points of a same triplet, 31 pixels for the size of the mask, 40degrees for the max angle between two consecutive tangents of a connected component, 7 pixels for the maximum distance between two consecutive points. These values were chosen to detect ellipses whose size is comprised between one half and one sixth of the sub-image. The following examples present the detected ellipse (or arc of ellipse) in several cases: different sizes, party hidden, different point of view.

Figure 12: Ellipses of different sizes and partly hidden

Figure 13: Three arcs of a unique ellipse

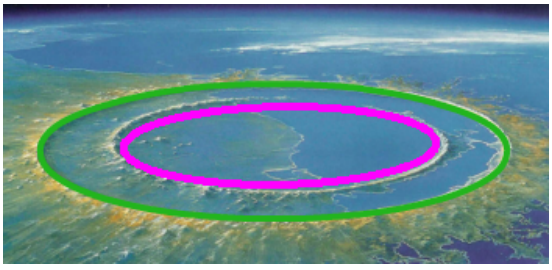


Figure 14: Two very flat ellipse with noisy edges

4.2 Exploring the limits of this algorithm

Several parameters can be studied to provide an extensive evaluation of the algorithm robustness; In particular, we can produce different types of noise, we can consider different sizes of arcs (in terms of ratio of the whole ellipse), different numbers and locations of very small arcs, ... but it should be too long and it is not the goal of this communication. So we decided to consider the variation of only a very few but relevant parameters in order to show how it works in limit cases.

In order to do that, we have simplified the protocol by directly starting with the binary image (we note that, in this case, the tangent line evaluation is less accurate than when using the grey level image).

We produce binary images that are based on the projection of an 3D circle viewed under a varying angle, with a given level of discretization (we do not visualize a polygon that fits an ellipse but a set of points distributed on this ellipse), and with a varying amount of noise.

On the next figures, we display images from our testing program. The first one (figure 14) is the original image; then (figure 15), we visualize the points supporting the primitives; and finally (figure 16), the Hough space shows a very good detection of the ellipse center.

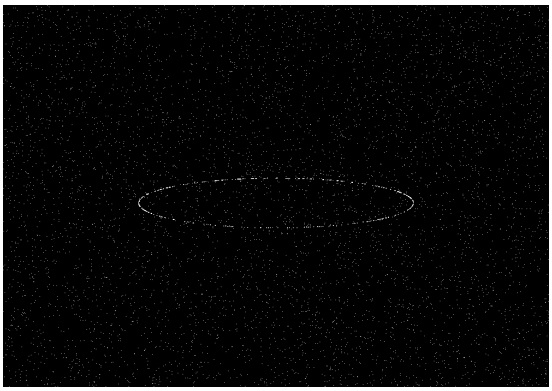


Figure 15: The original binary image (partly hidden ellipse and noise)

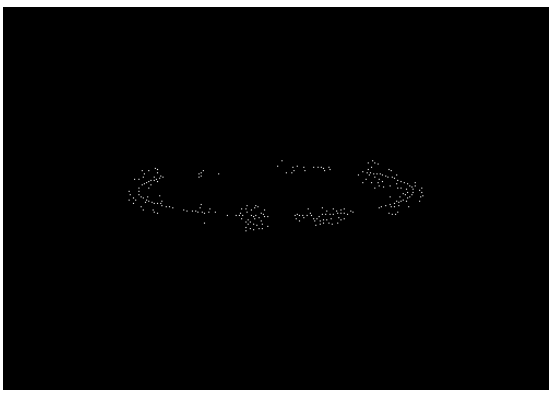


Figure 16: Points supporting primitives

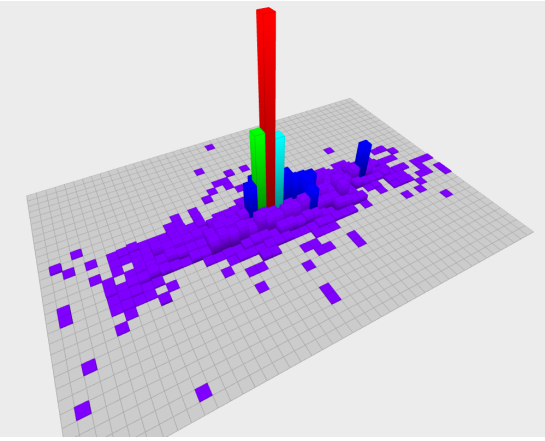


Figure 17: The corresponding Hough Space

The next figures provide a synthetic visualization of the robustness of the algorithm for various angles.

Each diagram is the result for different angles: in these cases, the angles are of 75 degrees, 80 degrees and 88 degrees (they measure the angle between an orthogonal vector to the ellipse plane and the viewing direction; e.g. in the last case – 88 degrees – it means that the viewpoint is almost in the plane of the ellipse – 2 degrees over it – and so the ellipse is very flat).

In each diagram, we put on the right (where it is written ”degradation”) the ratio of points of the ellipse that have been removed (0% means we fully display the ellipse, and 100% means we do not display any point of the ellipse). And we put on the left (where it is written ”Bruit”) the ratio of noise (this ratio is the number of points added randomly compared to the number of points of the fully displayed ellipse – e.g. 200 means there are twice more points). For example, at the cell ”Bruit = 300 and Degradation = 50”, we have only 50% of the points of the ellipse (50% have been randomly removed) and 6 times more points added randomly. These perturbations also deteriorates the tangent lines computed with the set of mask.

For each cell, we ran the detection program 1000 times (after 1000 times, the ratio is stabilized) with different sets of random values (under the constraints of ratios of the corresponding cell) and we used the following color code:

- green is for less than 10% of failure
- yellow is between 10% and 25% of failure
- orange is between 25% and 50% of failure
- red is for more than 50% of failure

These diagrams are very interesting because they drove us in the choice of the algorithm parameters. For example, we can see that for angles close to 90 degrees (flat ellipses); the sensitivity of the detection is more important for noise than for the number of points of the ellipse (with only 20% of points and a noise that is not too important, we detect such flat ellipses; thus, it should be better to threshold more drastically the grey-level image of contours in order to provide a better detection of flat ellipses.

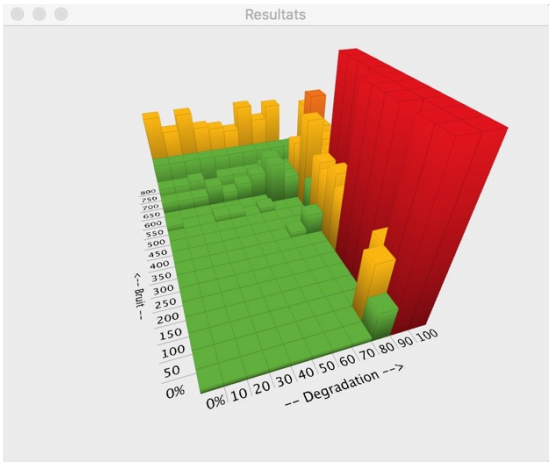


Figure 18: Angle value is 75 degrees

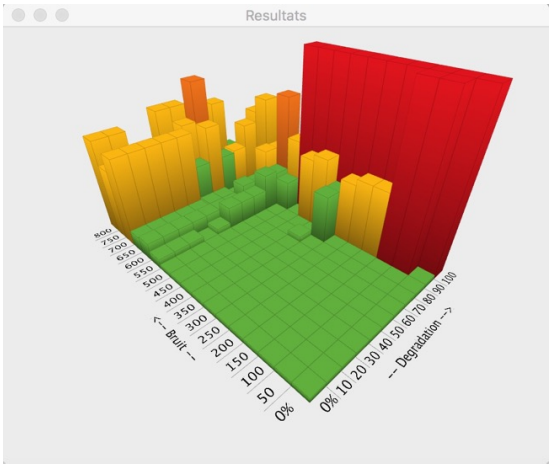


Figure 19: Angle value is 80 degrees

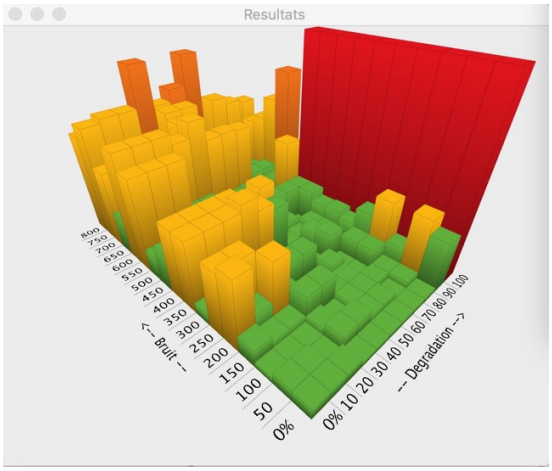


Figure 20: Angle value is 88 degrees

5 CONCLUSION

The algorithm proposed in this communication provides a solution to the ellipse detection problem when the ellipse is dashed, partly visible or very flat, all these situations being difficult to be solved with classical algorithms.

As it was mentioned before, this algorithm has to be used in complement to another one in order to avoid false negative, knowing that it does produces a very low amount of false positive (thanks to the validation step), when we need to have a complete and robust solution to ellipse detection.

6 REFERENCES

[Bas05] C. A. Basca, M. Talos and R. Brad, Randomized Hough Transform for Ellipse Detection with Result Clustering, EUROCON 2005 - The International Conference on "Computer as a Tool", Belgrade, 2005, pp. 1397-1400.

[Can86] J. Canny, A Computational Approach to Edge Detection, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8-6, pp. 679-698, 1986.

[Dud72] Duda, Richard O. and Hart, Peter E., Use of the Hough Transformation to Detect Lines and Curves in Pictures, in Commun. ACM, vol. 15-1, pp. 11-15, 1972.

[For14] Michele Fornaciari, Andrea Prati, Rita Cucchiara, A fast and effective ellipse detector for embedded vision applications, Pattern Recognition, Volume 47, Issue 11, November 2014, Pages 3693-3708, ISSN 0031-3203, http://dx.doi.org/10.1016/j.patcog.2014.05.012.

[Jia16] Qi Jia and Xin Fan and Zhongxuan Luo and Lianbo Song and Tie Qiu; A Fast Ellipse Detector Using Projective Invariant Pruning, in CoRR, vol. abs/1608.07470, 2016.

[Lib07] Libuda Lars and Grothues Ingo and Kraiss Karl-Friedrich, Ellipse Detection in Digital Image Data Using Geometric Features, in Advances in Computer Graphics and Computer Vision: International Conferences VISAPP and GRAPP 2006, Setúbal, Portugal, February 25-28, 2006.

[Lu15] T. Lu, W. Hu, C. Liu and D. Yang, "Effective ellipse detector with polygonal curve and likelihood ratio test," in IET Computer Vision, vol. 9, no. 6, pp. 914-925, 12 2015. doi: 10.1049/iet-cvi.2014.0347

[Pra13] Dilip K. Prasad, Maylor K.H. Leung, Chai Quek, ElliFit: An unconstrained, non-iterative, least squares based geometric Ellipse Fitting method, Pattern Recognition, Volume 46, Issue 5, May 2013, Pages 1449-1465, ISSN 0031-3203, http://dx.doi.org/10.1016/j.patcog.2012.11.007.

Improved Adaptive Background Subtraction Method Using Pixel-based Segmenter

Kemal Batuhan Baskurt
Department of Computer Engineering
06830, Ankara, Turkey
batuhanbaskurt@gmail.com

Refik Samet
Department of Computer Engineering
06830, Ankara, Turkey
samet@eng.ankara.edu.tr

ABSTRACT

Moving object detection is essential in many computer vision systems as it is generally first process which feeds following algorithmic steps after getting camera stream. Thus quality of moving object detection is crucial for success of the whole process flow. It has been studied in the literature over the last two decades but it is still challenging issue because of factors such as background complexity, illumination variations, noise, occlusion and run-time performance requirement considering rapidly increasing image size and quality. In this paper, we try to contribute to solve this problem by improving an existing real-time non-parametric moving object detection method. In scope of this study, pixel based background model in which each pixel is represented separately by its distribution on time domain is used. Mentioned discrete background model is suitable for parallel processing by dividing the image to sub images in order to accelerate the process. Main feature of proposed non-parametric approach is automatic adjustment of algorithm parameters according to changes on the scene. This feature provides easy adaptation to environmental change and robustness for different scenes with unique parameter initialization. Another contribution is scene change detector to handle sudden illumination changes and adopt the background model to new scene in the fastest way. Experiments on 2012 ChangeDetection.net dataset show that our approach outperforms most state-of-the-art methods. Improvement obtained both on robustness and practical performance provides our approach to be able to use in real world monitoring systems.

Keywords

Real-time systems, computer vision, video surveillance, video processing, moving object detection, background subtraction

1. INTRODUCTION

Rapid incensement on using surveillance cameras in daily life has resulted in the need to find effective methods and algorithms to overcome huge data gathered every second. Moving object detection is one of the most commonly used methods to give the meaning of the raw data. A popular approach to solve moving object detection problem is background subtraction which has been studied in the literature over the last two decades. The idea of background subtraction is calculating difference between current frame and the background model which represents the scene regarding to data obtained from previous frames. A complete background subtraction method has three main components. First one is background modelling which tries to represent the scene characteristic in best way. Second is subtraction operation which indicates the method to calculate difference between background model and current frame. Third one is background update mechanism that provides adaptation to scene changes.

Background subtraction is a challenging problem since background might include large image variations due to lightening, repetitive motions, crowded scene and occlusions. These environmental difficulties make the background modelling complex and time-consuming. Besides handling problems mentioned above, run-time performance of background subtraction method is also important considering high quality images gathered from surveillance cameras.

First approaches on background subtraction in literature focused on static background model. The model contains just one image of the scene. Each pixel of received image compared with related pixel on background image to determine if it belongs to the background. While this approach might useful for analyzing short video sequences in controlled environment, it cannot handle multiple backgrounds like waving flags or trees. Therefore researchers worked on more sophisticated statistical background models such as Gaussian Mixture Model [Sta99], codebook [Kim05] or [Kae02]. Other authors have worked on other approaches like using collected pixel values instead of generation statistical model [Wan07, Bar11, Van12 and Hof12]. While some approaches use low-level features such as color and texture [Zha06, Kri06 and Jia08], sub-pixel edge map [Jai07], Sobel edges [Aza10],

others try to solve problem using high-level semantic information of the scene on convolutional neural networks [Bra16]. There are wide scale surveys discussing theoretical backgrounds and evaluating run-time performance of background subtraction methods [Goy12, Sob14 and Vac12].

Our approach is based on PBAS method [Hof12], the differences lie in the neighbor update mechanism, automatic adjustment of increment/decrement of scene adaptation parameters and scene change detector algorithm for sudden illumination variations. Run-time performance of our approach is also improved by dividing the operation on sub-images under favor of discrete background model.

The paper is organized as follows. Section II describes related background subtraction methods. In section III, details of our approach are presented. Experiments and discussions are provided in section IV and section V concludes the paper.

2. RELATED WORK

Background subtraction methods aim to subtract moving object from static background without any priori information. Many methods have been proposed and extended survey papers can be found on this topic [Goy12, Sob14 and Vac12]. Existing methods can be divided into two groups as simple frame based methods and pixel based modelling methods.

Frame based methods also can be mentioned frame difference methods which use single image as background model. There are different approaches on building background image. Some approaches uses an image captured when there is no motion on the scene. Others simply calculate the difference between consecutive frames which means previous frame is always used as background model. [Lai98] describes background image by arithmetic mean of frames gathered at the training stage of their method. Absolute difference of background image and current image is used to determine motion area of the scene on all mentioned frame based approaches. They are unimodal approaches that background of each pixel is modelled by single value. These approaches are fast and easy to operate and efficient to detect instant motion on low-dynamic scenes. However frame based methods cannot handle dynamic background and long-term changes on the scene. Therefore more complex background models are proposed to solve environmental problems.

Over the years, several complex pixel level algorithms have been proposed. Most popular is Gaussian Mixture Model (GMM) [Sta99] which consists of modelling the distribution of the values observed over time at each pixel by a weighted mixture of Gaussians. This model handles most of the problems occurred because of the lack of multimodal background representation. Since its introduction, this model has been based by a lot of researchers and improved methods have been proposed. Main problem of GMM is high computational cost that prevents effective real-time operation of the method.

Another popular method is Codebook [Kim05] which models each pixel by a codebook which is a compressed form of background model. Each codebook contains codewords comprising colors transformed by an innovative color distortion metric. The method creates codebook model on training phase, then each frame is compared with the codebook model on test phase. Training phase of the method avoids adaptation to dynamic scene as significant changes on the scene after training cannot be handled. Necessity of training phase also makes the method inefficient in the meaning of easy-to-use. SACON [Wan07] method brought a new ‘non-parametric’ perspective by using collection of most recent image values at each pixel instead of statistical approach on background modelling. Each pixel of current frame is compared with stored collection of its previous values. Current pixel is labelled as background when it is 90% consistent with the background model. Oldest component of the background model is updated with new pixel value on update mechanism. ViBE [Bar11] and ViBE+ [Van12] use same background model with SACON but random component of the model is updated instead of the oldest one. Their decision criteria for background labelling are just 2 match with the background collection that makes the method faster than others. Mentioned fast update mechanism is built on conservative principle in which background model is only updated by background pixels.

Another non-parametric method PBAS [Hof12] also uses similar background model with ViBE but the randomness and decision thresholds are not fixed for all pixels as ViBE. Algorithm parameters are set separately for each pixel and they are changed dynamically according to scene variations over time. Mentioned dynamic parameter infrastructure makes the method more consistent regarding to scene changes while run-time performance decreases because of extra computational cost.

Meanwhile [Bra16] carries background subtraction on a different domain to solve the problem with spatial features learned with convolutional neural networks. Background model is generated by a single image and scene-specific training dataset. Their study indicates potential of deep features learned with conventional neural network for background subtraction without intention of proposing real-time and adaptive technique.

Our approach tries to improve deficiencies of PBAS method. Automatic adjustment of scene adaptation increment and decrement parameters which are used fixed in PBAS is added. Sudden illumination changes such as cloud passes, explosions caused by headlight or lightening variations on day-night change are important problems on real world applications. Background model is distorted by these artefacts and handling

the distortion takes time with normal update mechanism of the method. Our approach contains scene change detector algorithm to cope with this problem. Background model is updated in fastest way once sudden illumination change detected. Scene change detector provides stability of our approach against uncontrolled environmental changes on long term analysis. Neighbor update mechanism of PBAS is also changed in our approach to avoid possibility of adding foreground pixels to background model. Finally image is divided into sub-images and the algorithm is applied in parallel through discrete structure of background model considering sub-image borders on neighbor update mechanism.

3. PIXEL-BASED ADAPTIVE SEGMENTER

Background subtraction method used in our approach consists of five main steps as follows:

a) Background/foreground decision
b) Background update
c) Dynamic update of decision threshold
d) Dynamic update of background update rate
e) Scene change detector

Proposed approach uses background model proposed by [Hof12]. Our approach differs from the base model on update mechanism, parameter dynamism and completely new scene change detector.

First of all, initial background model is created in our approach. Each pixel of new video frame is compared with its background model in order to decide whether it is background or not. Then background model is updated if the current pixel is labelled as background. Decision threshold used in first step and background update frequency are dynamically adjusted according to scene variation. Finally scene change detector is used to handle sudden dominant changes that distorts reliability of background model. Technical details and contribution of our approach on each step are explained in this section.

3.1 Background/Foreground Decision

Background/foreground decision step aims to compare each pixel with its background model and decide whether it is background or foreground pixel. Background model of a pixel $B(x_i)$ represents *N* recently observed pixel values:

$$B(x_i) = \{B_1(x_i), B_2(x_i), \dots, B_N(x_i)\} \quad (1)$$

A pixel is labelled as background when it's current value ($I(x_i)$) is closer than decision threshold ($R(x_i)$) at least minimum number ($\sharp_{min}$) of the *N* background samples, as shown in Fig. 1. Thus decision threshold represents distance between current pixel's value and background samples in colour space of input image.

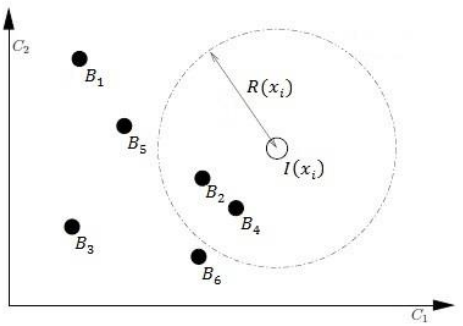


Figure 1. Background/foreground decision for 2-dimension (C_1, C_2) colour space

3.2 Background Update

Background model of a pixel is updated if it is labelled as background. Update operation is carried out by assigning current pixel value $I(x_i)$ to random selected sample $B_k(x_i)$ ($k \in 1, 2, \dots, N$). Current situation of the scene is learnt by background model in this way. Learning operation must be performed according to the scene change frequency. Thus, background model is updated in $p = 1/T(x_i)$ frequency instead of each frame. $T(x_i)$ represents pixel-based update rate which is adjusted dynamically with regarding to scene variation (see Section 3.4 for $T(x_i)$ definition).

Background model of random selected 8-connected neighbour of updated pixel ($y_i \in Neig(x_i)$) is also updated by neighbour's current pixel value in [Hof12]. Our approach proposes updating random selected neighbour's background with pixel value which is labelled as background (Eq. 2).

$$B_k(y_i) \leftarrow I(x_i) \quad (2)$$

Updating background model with the neighbour's pixel value is not appropriate as it may be labelled as foreground. Updating background model with foreground pixel value is prevented in our approach as it is inconsistent with principle of conservative background model.

3.3 Dynamic Update of Decision Threshold

Scene may have dynamic and stable regions at once, thus using fixed decision threshold and update rate for all pixels is insufficient considering real world scenarios. Decision threshold must be higher for dynamic regions that mean possibility of labelling moving pixels as background must be low. On the other hand, smaller changes on stable region must be considered for foreground with low decision threshold.

Minimum distance vector $D(x_i)$ between each updated background sample ($B(x_i)$) and current pixel value is stored to calculate pixel dynamism which provides to adjust decision threshold according to pixel changes on time domain.

$$D(x_i) = \{D_1(x_i), \dots, D_k(x_i), \dots, D_N(x_i)\} \quad (3)$$

$$d_{min}(x_i) = min_k dist\big(I(x_k), B_k(x_i)\big) \quad (4)$$

$$D_k(x_i) \leftarrow d_{min}(x_i) \quad (5)$$

Pixel dynamism is represented by average of minimum distance values for all background samples of the pixel. $R(x_i)$ is increased/decreased by increment/decrement parameter ($R_{inc/dec}$) when dynamism reaches to upper limit which is determined by R_{scale} parameter in Eq. 6.

$$R(x_i) = \begin{cases} R(x_i) \cdot (1 - R_{inc/dec}), if\ R(x_i) > \bar{d}_{min}(x_i) \cdot R_{scale} \\ R(x_i) \cdot (1 + R_{inc/dec}), \quad else \end{cases} \quad (6)$$

$R(x_i)$ is limited by lower decision value (R_{lower}) in order to control decision criteria in acceptable limits.

3.4 Dynamic Update of Background Update Rate

Background model update rate $T(x_i)$ which represents update frequency (in frames) of the model is another pixel-based adaptive parameter related to pixel dynamism. Background model of high-dynamic region is updated rare than stable region for preserving background model from moving objects. Update rate of foreground pixel is increased for rare update on the region. $1/T(x_i)$ is update frequency where $T(x_i)$ is calculated as follows:

$$T(x_i) = \begin{cases} T(x_i) + \dfrac{T_{inc}}{\bar{d}_{min}(x_i)}, & if\ F(x_i) = 1 \\ T(x_i) - \dfrac{T_{dec}}{\bar{d}_{min}(x_i)}, & if\ F(x_i) = 0 \end{cases} \quad (7)$$

where ($F(x_i) = 1$) represents foreground pixel, while ($F(x_i) = 0$) represents background. $T(x_i)$ parameter is limited between minimum (T_{lower}) and maximum (T_{upper}) values in order to control the background model in the case of false updating.

Increment/decrement parameters of both decision threshold ($R_{inc/dec}$) and update rate (T_{inc}, T_{dec}) are also adjusted dynamically in our approach while they are fixed in [Hof12]. Using fixed increment/decrement step for all pixels during entire run-time causes slow reaction of the method over fast changes in the scene. Our approach on this point accomplishes complete adaptation to dynamic scene. Each pixel uses its own increment/decrement parameter instead of unique ones for all. Increment/decrement parameters are changed in 1 % ratio according to dynamism of the pixel for each step. Mentioned parameters are increased for stable pixels while decreased for dynamic pixels. In the case of sudden change on stable region, threshold parameters are quickly adopted to new scene because of bigger increment/decrement steps.

$$R_{inc/dec}(x_i) = \begin{cases} R_{inc/dec}(x_i) \cdot 0.99, if\ R(x_i) > \bar{d}_{min}(x_i) \cdot R_{scale} \\ R_{inc/dec}(x_i) \cdot 1.01, \quad else \end{cases} \quad (8)$$

$$T_{inc}(x_i) = \begin{cases} T_{inc}(x_i).0.99, & if\ F(x_i) = 1 \\ T_{inc}(x_i).1.01, & if\ F(x_i) = 0 \end{cases} \quad (9)$$

$$T_{dec}(x_i) = \begin{cases} T_{dec}(x_i).0.99, & if\ F(x_i) = 1 \\ T_{dec}(x_i).1.01, & if\ F(x_i) = 0 \end{cases} \quad (10)$$

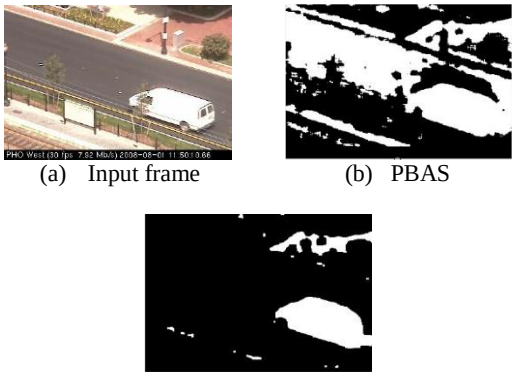
3.5 Scene Change Detector

Dominant changes on the scene such as streetlight (de)activation on day/night change, cloud passing or photocell lightening cause serious problem for background subtraction methods even dynamic update rate is used. Normal adaptation process of background model to new scene structure takes long time which means a large number of false detection during this period. Our approach has a precaution named scene change detector for handling this unusual situation. Maximum motion ratio M_{Max} is defined to control unexpected dominant changes.

Update rate of foreground pixels is assigned to T_{lower} which means highest update frequency when the ratio of foreground pixels reaches to M_{Max}. Update ratio resumes to normal frequency, after then background model learns the scene in the fastest way.

$$T(x_i) = \begin{cases} T_{lower}, & \#_{(F=1)} > (M_{Max} \cdot \#_x) \\ T(x_i), & else \end{cases}, i \in (F(x_i) = 1) \quad (10)$$

Effect of scene change detector compared to PBAS result is showed in Fig 2.



(a) Input frame (b) PBAS

(c) Our Approach

Figure 2. Effect of scene change detector (818th frame of boulevard scenario on camera jitter category of change detection dataset)

Our approach is more adaptive than based approach proposed by [Hof12] considering dynamic adjustment of more parameters. Fewer parameters adjusted by user makes our approach more robust against different scenes. Scene change detector also solves common problem faced in real world applications.

4. IMPLEMENTATION AND EXPERIMENTAL RESULTS

Following metrics are used to evaluate performance of proposed approach (Table 1).

Metric	Explanation
Recall	TP/ (TP + FN)
F1	(2 * Precision * Recall) / (Precision + Recall)
Precision	TP / (TP + FP)

Table 1. Performance Metrics (TP: True Positive, FP: False Positive, FN: False Negative, TN: True Negative)

Recall represents fraction of number of foreground pixels classified as foreground over number of foreground pixels classified as background. Precision represents fraction of number of foreground pixels classified as foreground over number of background pixels classified as foreground. F1 represents harmonic average of recall and precision.

Our approach is compared to PBAS (implementation provided by the authors) in terms of three performance measures (Table 2) on six scenarios provided by change detection 2012 benchmark [Goy12].

Run-time performance comparison of both methods is also provided in Table 2 (bold values are the best in the comparison). Change detection 2012 benchmark provides extensive comparison of 44 state-of-the-art methods including PBAS. Thus comparing our approach with PBAS on this dataset also provides opportunity to evaluate our performance among other state-of-the-art methods.

Optimal parameter setting proposed by the authors of PBAS is follows: {N = 35, $\#_{min}$ = 2, $R_{inc/dec}$ = 0.05, R_{lower} = 18, R_{scale} = 5, T_{inc} = 1, T_{dec} = 0.05, T_{lower} = 2, T_{upper} = 200}. Same parameter setting except T_{upper} = 5 is used for our approach. As mentioned in previous section $R_{inc/dec}$, T_{inc} and T_{dec} parameters are adjusted dynamically according to scene variation in our method. Thus defined values above are initial ones for these parameters. They are changed in the ratio of 0.01 according to dynamism of related pixel. Limitation parameters for these ones are used as follows: {$R_{inc/dec}^{upper}$ = 0.05, $R_{inc/dec}^{lower}$ = 0.01, T_{inc}^{upper} = 1.5, T_{inc}^{lower} = 0.5, T_{dec}^{upper} = 0.1, T_{dec}^{lower} = 0.02}

Result of each scenario with the overall of each performance measure is provided. Our approach shows better performance in all measures of overall.

As mentioned in Section III, discrete background model in which each pixel is represented separately by its recently obtained values is used in our approach. Discrete model provides opportunity of parallel processing by dividing scene to sub images and operate them separately. Important point to pay attention on parallelization is updating background model of neighbor pixel on intersection region of sub-images. Border control is added to avoid manipulation of others memory between threads. Our implementation divides the scene 2x2 sub-images for parallel processing. Run-time performance presented in Table 2 shows effectiveness of our implementation. Our approach is processed 61 % faster than PBAS on overall.

5. CONCLUSION

We have presented improvement of our approach across PBAS. Three more parameters are adjusted dynamically according to scene change instead of using constant values. Neighbour update mechanism is changed to prevent updating background model with foreground pixel values. Moreover our scene change detector algorithm provides fast adaptation to major changes on the scene without large number of false detection. Our approach also benefits from discrete structure of background model in order to parallelise the method on sub-images. Mentioned improvements both on algorithm and implementation outperform PBAS and most of state-of-the-art methods. Future work will focus on completely dynamic method without necessity of any constant parameter. Performance on intermittent object detection scenario which dramatically decreases the overall performance also seems to need improvement

		Baseline	Camera Jitter	Dynamic Background	Intermittent Object Motion	Shadow	Thermal	Overall
Recall	PBAS	**0,8259**	**0,7927**	**0,7918**	0,4409	0,8447	0,6395	0,7226
	Our Approach	0,8246	0,7114	0,7836	**0,5238**	**0,8665**	**0,6617**	**0,7286**
F1	PBAS	0,7820	0,5490	0,6183	0,4922	**0,7772**	0,6806	0,6499
	Our Approach	**0,8198**	**0,5617**	**0,6974**	**0,5379**	0,7550	**0,7176**	**0,6816**
Precision	PBAS	0,7698	0,4386	0,6535	**0,7206**	**0,7283**	0,7884	0,6832
	Our Approach	**0,8225**	**0,5211**	**0,7182**	0,6308	0,7130	**0,8176**	**0,7039**
Run Time (ms)	PBAS	25,25	33,50	32,00	16,50	23,67	17,00	24,65
	Our Approach	**19,75**	**18,5**	**17,33**	**10,5**	**15**	**10**	**15,18**

Table 2. Results of PBAS on all Scenarios of Change Detection Dataset

6. REFERENCES

[Aza10] Azab, M. M., Shedeed, H. A., & Hussein, A. S. "A new technique for background modeling and subtraction for motion detection in real-time videos, in Image Processing (ICIP), 2010 17th IEEE International Conference on (pp. 3453-3456). IEEE.

[Bar11] Barnich, O., & Van Droogenbroeck, M. "ViBe: A universal background subtraction algorithm for video sequences," Image Processing, IEEE Transactions on, 20(6), 1709-1724, 2011.

[Bra16] Braham, M., & Van Droogenbroeck, M. "Deep Background Subtraction with Scene-Specific Convolutional Neural Networks," in International Conference on Systems, Signals and Image Processing, Bratislava 2016. IEEE.

[Goy12] Goyette, N., Jodoin, P. M., Porikli, F., Konrad, J., & Ishwar, P. "Changedetection. net: A new change detection benchmark dataset," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on (pp. 1-8). IEEE.

[Hof12] Hofmann, M., Tiefenbacher, P., & Rigoll, G. "Background segmentation with feedback: The pixel-based adaptive segmenter," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on (pp. 38-43).

[Jai07] Jain, V., Kimia, B. B., & Mundy, J. L. "Background modeling based on subpixel edges," in Image Processing, 2007. ICIP 2007. IEEE International Conference on (Vol. 6, pp. VI-321). IEEE.

[Jia08] Jian, X., Xiao-qing, D., Sheng-jin, W., & You-shou, W. "Background subtraction based on a combination of texture, color and intensity," in Signal Processing, 2008. ICSP 2008. 9th International Conference on (pp. 1400-1405). IEEE.

[Kae02] KaewTraKulPong, P., & Bowden, R. "An improved adaptive background mixture model for real-time tracking with shadow detection," in Video-based surveillance systems (pp. 135-144). Springer US, 2002.

[Kim05] Kim, K., Chalidabhongse, T. H., Harwood, D., & Davis, L. "Real-time foreground–background segmentation using codebook model," Real-time imaging, 11(3), 172-185, 2005.

[Kri06] Kristensen, F., Nilsson, P., & Öwall, V. "Background segmentation beyond RGB," in Computer Vision–ACCV 2006 (pp. 602-612). Springer Berlin Heidelberg.

[Lai98] Lai, A. H., & Yung, N. H. "A fast and accurate scoreboard algorithm for estimating stationary backgrounds in an image sequence," in Circuits and Systems, 1998. ISCAS'98. Proceedings of the 1998 IEEE International Symposium on (Vol. 4, pp. 241-244). IEEE.

[Sob14] Sobral, A., & Vacavant, A. "A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos," Computer Vision and Image Understanding, 122, 4-21, 2014

[Sta99] Stauffer, C., Grimson, W. E. L. "Adaptive background mixture models for real-time tracking," in Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on. (Vol. 2). IEEE.

[Vac12] Vacavant, A., Chateau, T., Wilhelm, A., & Lequièvre, L. "A benchmark dataset for outdoor foreground/background extraction," in Computer Vision-ACCV 2012 Workshops (pp. 291-300). Springer Berlin Heidelberg.

[Van12] Van Droogenbroeck, M., & Paquot, O. "Background subtraction: Experiments and improvements for ViBe." in Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on (pp. 32-37).

[Wan07] Wang, H., & Suter, D. "A consensus-based method for tracking: Modelling background scenario and foreground appearance," Pattern Recognition, 40(3), 1091-1105, 2007.

[Zha06] Zhang, H., & Xu, D. "Fusing color and texture features for background model," in Fuzzy Systems and Knowledge Discovery: Third International Conference, FSKD 2006, Xi'an, China, September 24-28, 2006. Proceedings (pp. 887-893). Springer Berlin Heidelberg.

Fashion Recommendations Using Text Mining and Multiple Content Attributes

Wei Zhou, Yanghong Zhou, Runze Li and P.Y. MOK*
Institute of Textiles and Clothing
The Hong Kong Polytechnic University
Hunghom, Hong Kong
tracy.mok@polyu.edu.hk

ABSTRACT

Many online stores actively recommend commodities to users for facilitating easy product selection and increasing product exposure. Typical approach is by collaborative filtering, namely recommending the products based on their popularity, assuming that users may buy the products that many others have purchased. However, fashion recommendation is different from other product recommendations, because people may not like to go with the crowd in selecting fashion items. Other approaches of fashion recommendations include providing suggestions based on users' purchase or browsing history. This is mainly done by searching similar products using commodities' tags. Yet, the accuracy of tag-based recommendations may be limited due to ambiguous text expression and nonstandard tag names for fashion items. In this paper we collect a large fashion clothing dataset from different online stores. We develop a fashion keyword library by statistical natural language processing, and then we formulate an algorithm to automatically label fashion product attributes according to the defined library by text mining and semantic analysis. Lastly, we develop novel fashion recommendation models to select similar and mix-and-match products by integrating text-based product attributes and image extracted features. We evaluate the effectiveness of our approach by experiment over real datasets.

Keywords

Fashion recommendation, Text mining, Mix-and-match

1 INTRODUCTION

Due to the continued growth in the acceptance of e-commerce, the clothing products selling online has a rapid increase in the past two decades, not only in terms of sales volume but also in term of product types [1]. Recommendation technologies suggest users with products by analysing users' interests and purchase behaviors [2]. Content-based methods recommend items based on a comparison between the content of the items and a user's profile [3]. Recommender systems are widely used in e-commerce websites, such as amazon.com, netflix.com, taobao.com, and zalora.com.

Unlike traditional recommendation methods that focus on similarity search, we propose in this paper a method that makes recommendations in two dimensions, suggesting similar and mix-and-match clothing items from a content-based approach. In similar clothing recommendations, products that are similar to the target product are recommended to users; it helps users to quickly search for their needed products. For example, assuming the target product is a dress, the recommender system will suggest users other dresses that are similar to the given dress in colour, brand, description, and/or price range. All recommended products belong to the same product category, i.e., dress in this example. In mix-and-match clothing recommendation, the system gives outfit ideas that users are suggested with other clothing items or accessories that match well with the given product; and users can choose to buy these products together. Taking the dress example above, recommender system will suggest users with other products (e.g. jacket, handbag, shoes) that match well with the target dress. These matching products may match the target product in brand, colour, size, and/or style; all matching products belongs to categories different from the specific category of the target product.

The main contribution of this paper is that we propose a recommendation system that provide users with similar and mix-and-match clothing product recommendations based on a selected item. The rest of the paper is organized as follows. In the next section, we will look at previous work in the related areas. In Section 3,

we will discuss the overall structure and detail of the proposed recommendation system. Section 4 describes the detailed metrics used in the experiments and experimental methodology for the proposed methods. In Section 5, we will present our experimental results and relevant discussion.

2 RELATED WORK

With the rapid development of computer and internet technologies, the key problem has been changed from how to obtain the required information in the past to how to help users get the necessary knowledge from the massive information today. There are two branches about clothing product recommendations, similar clothing recommendation and mix-and-match clothing recommendation. In the literature, methods for computing similarity can be classified as semantic-based methods and case-based reasoning methods [7]. The existing methods calculate the similarity of different clothing items by extracting features from these clothing, evaluating the similarity of different features and summarising all features with a weighted average score [8]. There are a number of measures calculating the feature similarity, such as Euclidean Distance, Manhattan Distance, Cosine Similarity, Hamming distance, and Correlation coefficient [9]. Therefore, recommendation of similar clothing has turned out to be a problem of feature extraction from clothing products, including modeling and formulation. In [10], similarity among clothing items is measured based on the semantic description of clothing products, in which a lightweight fashion ontology is developed based on expert knowledge. In [11], a method is proposed to recognize clothing attributes from clothing images, but only upper-body clothes are analysed. It is found that image backgrounds affect the final results of clothing attribute recognition, and the recognition accuracy is difficult to ensure [12].

In mix-and-match recommendations, [13] proposed to calculate the compatibility of clothing items and attributes. Inter-object or inter attribute compatibility are considered, and Conditional Random Field (*CRF*) is used to calculate the most compatible combinations. Most reported studies on mix-and-match recommendations are content-based techniques using prior knowledge. Some researchers design mix-and-match rules based on experts who are familiar with clothing coordination and styling, but users' personal requirements are not considered [14]. Other researchers provide recommendations by mining association rules from massive transaction records, finding the frequent mode of clothing matches [15]. This method can capture the fashion trends of clothing but can not give professional clothing matches.

There are also some work on context-based personalized clothing recommendations. Wearing occasion is one of the most important factors to consider when people select clothes.

3 METHOD

In this paper, clothing recommendations for both similar products and mix-and-match products are proposed. All tests are based on real dataset crawled from online fashion websites. We consider the description of clothing products, using *Natural Language Processing* to extract keywords and features of clothing products. Then products are re-grouped and groups are encoded, match rules are defined by fashion stylists and clothing experts. On the other hand, a domain ontology about clothing is constructed. Based on this, product similarity is calculated and sorted. Finally, recommendations are made, including similar products and mix-and-match products.

We will describe three parts of development in this section. In Section 3.1, the structure of the proposed recommendation system is described, which follows with dataset used in this study (Section 3.2) and our clothing product data re-categorization (Section 3.3). In Section 3.4 introduces how similar recommendations are computed. In Section 3.5, the mix-and-match clothing product recommendation is explained.

3.1 Recommendation Structure

There are four modules in the recommendation system, including attribute extraction module, clothing product recommendation module, sorting module and user feedback module.

The first phase is attribute extraction. In this phase, description of clothing products are analyzed by *NLP* tools. The second phase is the recommendation module. Similar recommendation and mix-and-match recommendation are put forward in this phase. The third phase is the sorting module, the similarity score and mix-and-match score are calculated between the target product and all candidate clothing products. The result is sorted and clothing products with the highest scores will be selected in to the final recommendation list. Lastly, a method is used to improve the performance of the structure using user feedback. The ontology library and keyword extraction method will be adjusted according to the feedback. The proposed fashion recommendation structure is shown in Figure 1. In the flowing subsections, similar clothing product recommendation and mix-and-match clothing product recommendation will be introduced in details.

3.2 About the Dataset

The data used in this paper are crawled from online fashion websites, including zalora.com.hk, hm.com/hk

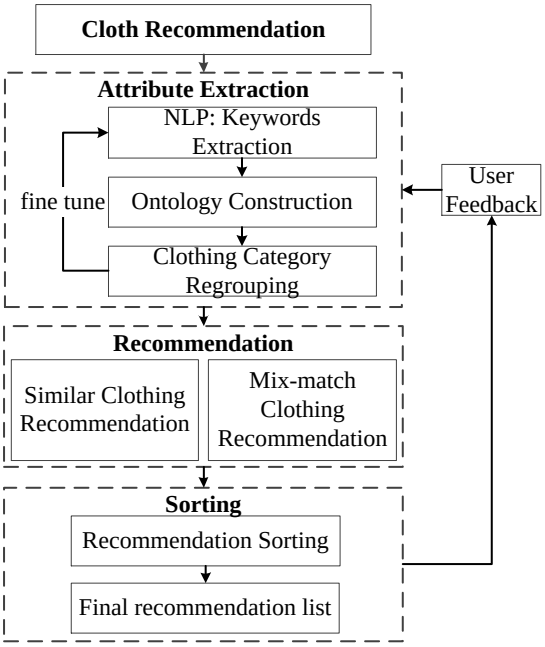


Figure 1: The structure of proposed recommendation method

and asos.com. There are a total of 158,211 products crawled from these websites, among which 91,491 are women products and 66,720 are men products. The main features we have got from online websites including product breadcrumbs, crawled time, product brand, product gender, product title, product price, product description, product details. The breadcrumbs of clothing products from different websites are different. In order to recommend products across websites and unify the breadcrumbs of all clothing products, we propose a product re-categorization method in Section 3.3 to map all products from the old breadcrumbs to our new redefined breadcrumbs.

3.3 Clothing Product Re-categorization

Since we crawled product data from different websites, the product classification vary among these websites. We therefore need to map products from the old categories to our new organized categories.

There are three kind of information we can use: clothing description, transaction data, clothing image. Description of clothing products is usually text information. In description of clothing products, including clothing category, which the product belongs to, product colour, price, materials. We use *NLTK* to extract nouns keywords from text description. *NLTK* is a leading platform for building Python programs to work with human language data. Transaction data are the records that users buying the products, including user name, product name, transaction data. A lot of information can be obtained from clothing images. Using pattern recognition techniques, information including colour, clothing type, clothing style can be mined from clothing images, which can make up the lack of text description of clothing products.

According to clothing products data we have got, we organize all clothing products into 101 categories, including 42 men's categories and women's 59 categories. Each category is assigned a unique identification code. Women's clothing category codes start with 'P3' and men's clothing category codes start with 'P2'. Table 3.3 shows the examples of keywords dictionary and the keywords list mapping that category.

For example, category code of bags (for women) is 'P302', and category code of bags (for men) is 'P202'. For each category, a list of keywords is found, if the keyword exists in one clothing description, the cloth product is assigned as the new category. For example, there is a bag(for women) product, if the text description of the product contains keywords in the keywords list of Category P302, the product is assigned as 'P302'. But there exist misclassified situation. The text description of clothing product is divided into three parts, product title, product breadcrumbs, product description. For each part, if the keywords list hit the text segment, the record will be recorded and scored. Category with the highest score would be selected.

A method is used to adjust and fine tune the keywords list of each category. The final result shows that more than 95% of the products are correctly categorized. After clothing product re-categorization, all clothing products are recategorized to 101 new categories. For each new category, a list of features are selected, which is also coded, features are selected and assigned to each product according to the text description of the product. Similar product recommendation and mix-and-match recommendation will be on basis of the new categories and features.

3.4 Similar Clothing Recommendation

3.4.1 Similarity Calculation Metrics

Similar clothing recommendation selects the most similar products in the same category. There are a list of features in each clothing product. Similarity calculation metrics of features can be represented as a vector,

$$C = \{f_1, f_2, f_3 \cdots f_m\} \quad (1)$$

where m is the number of features. f_i is the ith feature of the feature list. The main content of similarity computation is feature similarity computing and weight coefficient. There are three kinds of feature similarity computation, for explicit and continuous features, discrete features, and boolean features.

Category Name	Category Code	Keywords List
Bags	P302	Bags;Bag;Backpacks;Backpack;Clutches;Clutch;Handbag;Handbags;
Beachwear	P30401	Beachwear;Bikinis;Swimsuits;Bikini;Swimsuit
Dresses	P30402	Dresses;Dress
Pants_Leggings	P30406	Jobbers;Jobber;Leggings;Legging;Pants;Pant;Trousers;Trouser;Sweatpants
Bags	P202	Backpacks;Backpack;Bags;Bag;Clutches;Clutch;Duffle,Bags;Duffle Bag;Handbags
Beachwear	P20401	Beachwear;Bikinis;Bikini;Swimsuits;Swimsuit
Pants_Joggers	P20406	Chinos;Chino;Cropped,Pants;Cropped Pant;Cropped Trousers;Cropped,Trouser
Poloshirts	P20407	Poloshirts;Poloshirt;Polo,Shirts;Polo shirt;Polo;Long,Sleeved;Long-sleeved; Long-sleeves;Long-sleeve;Long sleeves;Long sleeve
Underwear	P20411	Boxers;Boxer;Bras;Bra;Briefs;Brief;Hipsters;Hipster;Lingerie,Sets;Lingerie Set; Lingerie;Panties;Pantie;Underwear

Table 1: Category re-categorization: examples of keywords dictionary

The similarity of explicit and continuous features can be computed with the nearest neighbor algorithm by Equation 2.

$$Sim(f_i, f_j) = 1 - \frac{|f_i - f_j|}{\beta - \alpha} \tag{2}$$

where $f_i \in [\alpha, \beta]$ and $f_j \in [\alpha, \beta]$ are the corresponding features in two clothing products. α is the minimum value and β is the maximum value in the corresponding clothing feature set.

The similarity of discrete features can be calculated using fuzzy mathematics:

$$Sim(f_i, f_j) = \frac{f_i \cap f_j}{f_i \cup f_j} = \frac{f_i \cap f_j}{f_i + f_j - f_i \cap f_j} \tag{3}$$

The similarity of boolean features can be either 1 or 0, and the similarity can be calculated as:

$$Sim(f_i, f_j) = \begin{cases} 1, f_i = f_j \\ 0, f_i \neq f_j \end{cases} \tag{4}$$

Lastly, the product similarity is calculated by integrating all available features:

$$Sim(C_i, C_j) = \frac{\sum_{i=1}^{m} w_i Sim\left(C_{f_i}, C_{f_j}\right)}{\sum_{i=1}^{m} w_i} \tag{5}$$

where C_i and C_j are two feature vectors of clothing products, w_i is the weight for the ith feature, while $Sim\left(C_{f_i}, C_{f_j}\right)$ is the similarity of ith feature of clothing product C_i and C_j. A method combining nearest neighbor method and fuzzy similarity retrieval method is proposed in this paper.

3.4.2 Color Similarity Calculating

One of the goals of the similar product recommendation is to calculate the similarity of product colors. Color is one of the most difficult areas to normalize. In fact, one single colour has different possible descriptions. However, it is difficult to calculate the similarity of these descriptions. These colours can be understood by people, but it is very difficult for computers. Even simple colours such as *snakeskin* or *periwinkle* can be difficult to handle automatically. In this paper, a method is proposed to cluster colours into a list of color categories. After cleaning all the color descriptions, there are 6,700 different colour descriptions in the dataset. All these colour descriptions are mapped into 11 categories.

3.5 Mix-and-match Clothing Recommendation

As mentioned in Section 3.3, we have 42 men categories and 59 women categories; each category has various features. In this section, mix-and-match rules are defined by fashion expert. Women's and men's mix-and-match are set respectively. Two match levels are defined, category level and feature level. A mix-and-match matrix is constructed for women's categories and men's categories. For example, a 59×59 matrix is constructed, if there exist match possibility in two categories, there would be a none zero score in the cross position of the mix-and-match matrix. The scores are divided into six levels,0, 0.2, 0.5, 0.8, 1 and 3 respectively, which 0 means the two categories do not match, while 1 means two categories match well. Score 3 means there exits feature level mix-and-match. Feature level match is also defined by fashion expert.

4 EXPERIMENT

4.1 Experiment Setup

In this section, the proposed method is tested on real online dataset by experiment. A total of 50 human subjects, mainly undergraduate students of The Hong Kong Polytechnic University participated in the experiment. The experiment is set as follows: 200 products are randomly selected involving various product categories. These products are divided into two groups: Group I with 50 products that computed recommendations are not shown, and Group II with 150 products that computed recommendations are shown

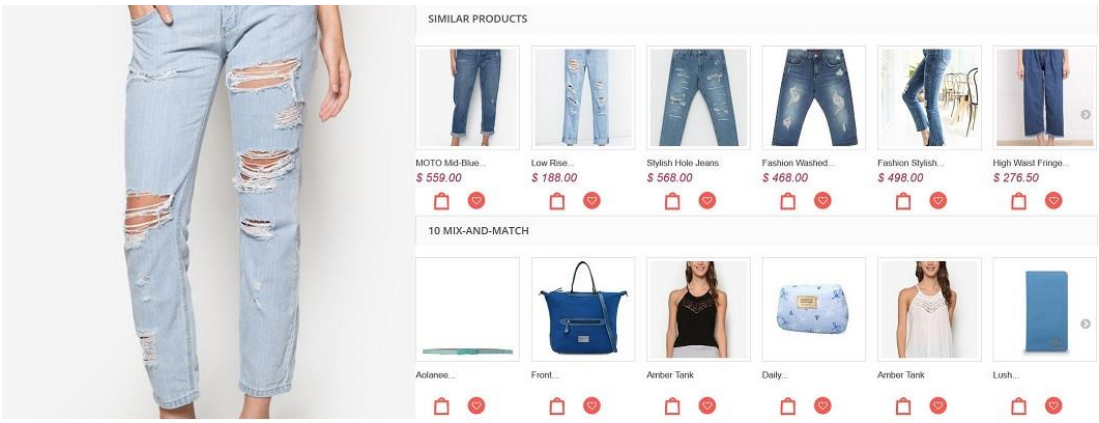


Figure 2: An example of similar and mix-and-match recommendation

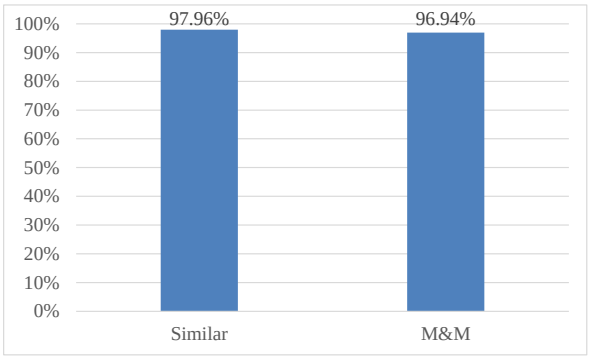


Figure 3: Similar products and mix-and-match recommendations volunteers selected

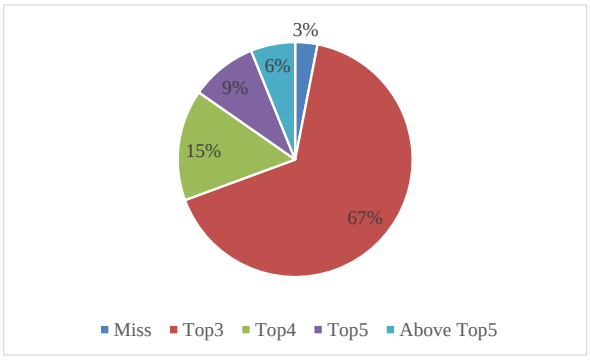


Figure 4: Distribution of products volunteers selected *VS* similar products recommendations

to the subjects. Subjects were asked to recommend 3 similar products and 4 mix-and-match products for each product of Groups I and II.

4.2 Experimental Results

There are 4820 product recommendations results in the experiment, during the period from 6th Feb to 18th Feb 2017. After data cleaning, there are 1323 records, including 441 Group I and 882 Group II ratings.

In Group I, subjects were asked to choose 3 similar products and 4 mix-and-match products without any recommendation suggestions. In Group II, subjects were asked to choose 3 similar products and 4 mix-and-match products from the recommendation list. If what subjects choose on their owns are happened same as those suggested by the recommendation system, it is an indicator that the system can generate very effective recommendations that are comparable to humans.

We judge the recommendation effectiveness by the sequence of what the users have chosen. Figure 3 shows that most users can select similar and mix-and-match items from the recommendation lists, which the hit rates are 97.96% and 96.94% for similar and mix-and-match items, respectively. Figure 4 shows which products users selected from the similar recommendation list, which are compared to the system calculated most similar items. We randomized the recommendations before showing to the subjects, and what the subjects selected are compared to the score (score sorting the similarity of the products by the system). If users selected items are the most similar ones among the randomized recommendations, it means the similar recommendations are effective. The results showed that over 67% of users selected similar items are the top 3 recommended items. Considering that users are asked to select 3 most similar items form the list, the similar recommendation is quite successful. Top 4 and top 5 rates are 15% and 9% respectively, meaning the users selected 3 items are among the top 4 most similar and top 5 most similar items in the recommendation list. Only 3% of selected items are not recommended by the system. Figure 5 shows the mix-and-match recommendations, which is more challenging problem as people are subjective about what products mean to be good match with the given items. The results show that less than half of the users selected mix-and-match are top 4 best match items suggested by the system. Considering that users are required to pick 4 items, it means 100% hit rate

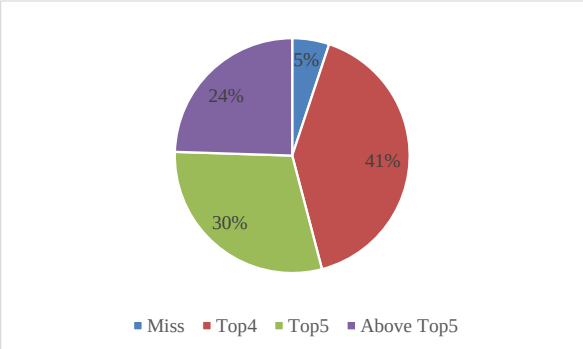


Figure 5: Distribution of products volunteers selected *VS* mix-and-match products recommendations

for these top 4 hit rate. Apart from full hit (top 4), around 30% of users selected mix-and-match items are among the top 5 suggestions of the system. Only 5% of the mix-and-match are not on the system generated recommendations, but some random 'noise' items given by the system. The results of mix-and-match hit rates are quite promising. Two examples of similar and mix-and-match recommendation is shown in Figure 2. More instances can be found in http://www.tozmart.com/.

5 CONCLUSIONS AND FUTURE WORK

In this paper, a hybrid recommendation structure on fashion clothing products is proposed, including similar and mix-and-match recommendations. Results show that the method get a good result both in similar recommendations and mix-and-match recommendations. In this paper, images of fashion products are not fully used, however many useful information for product recommendation cannot be extracted from text descriptions. In the future, we will focus on extracting features from clothing product images. A deep learning framework will be used to parse fashion photos, after which we will extract multi-features from parsed regions to describe product attributes like pattern, colour and style details. On the other hand, user context information and personalized recommendation will also be considered.

6 ACKNOWLEDGMENTS

The work described in this paper was partially supported by the Innovation and Technology Commission of Hong Kong under grant ITS/253/15, and a grant of Guangdong Provincial Department of Science and Technology under the grant R2015A015. Wei Zhou was also supported by National Natural Science Foundation of China (Under grant No. 61602070).

7 REFERENCES

[1] Frejlichowski D, Czapiewski P, Hofman R. Finding Similar Clothes Based on Semantic Description for the Purpose of Fashion Recommender System. Asian Conference on Intelligent Information and Database Systems[C]. Springer Berlin Heidelberg, 2016: 13-22.

[2] Di W, Wah C, Bhardwaj A, et al. Style finder: Fine-grained clothing style detection and retrieval[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2013: 8-13.

[3] Liu S, Feng J, Song Z, et al. Hi, magic closet, tell me what to wear![C]. Proceedings of the 20th ACM international conference on Multimedia. ACM, 2012: 619-628.

[4] Yamaguchi K, Okatani T, Sudo K, et al. Mix and Match: Joint Model for Clothing and Attribute Recognition[C]. BMVC. 2015: 51.1-51.12.

[5] Liu Z, Wang J, Chen Q, et al. Clothing similarity computation based on TLAC[J]. International Journal of Clothing Science and Technology, 2012, 24(4): 273-286.

[6] de Melo E V, Nogueira E A, Guliato D. Content-based filtering enhanced by human visual attention applied to clothing recommendation[C]//Tools with Artificial Intelligence (ICTAI), 2015 IEEE 27th International Conference on. IEEE, 2015: 644-651.

[7] Dai W. Clothing Fashion Style Recommendation System[D]. Northeastern University Boston, 2015.

[8] Long S, Zhu W. Mining evolving association rules for e-business recommendation[J]. Journal of Shanghai Jiaotong University (Science), 2012, 17(2): 161-165.

[9] Yu-Chu L, Kawakita Y, Suzuki E, et al. Personalized clothing-recommendation system based on a modified Bayesian network[C]//Applications and the Internet (SAINT), 2012 IEEE/IPSJ 12th International Symposium on. IEEE, 2012: 414-417.

[10] Dong J, Chen Q, Feng J, et al. Looking inside category: subcategory-aware object recognition[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2015, 25(8): 1322-1334.

[11] Yamaguchi K, Kiapour M H, Ortiz L E, et al. Retrieving similar styles to parse clothing[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(5): 1028-1040.

[12] Plumbaum, T. and B. Kille (2015).Plumbaum T, Kille B. Personalized fashion advice[M]//Smart Information Systems. Springer International Publishing, 2015: 213-237.

[13] Lee H, Lee S. Style Recommendation for Fashion Items using Heterogeneous Information Network[C]//RecSys Posters. 2015.

[14] Gomaa W H, Fahmy A A. A survey of text similarity approaches[J]. International Journal of Computer Applications, 2013, 68(13).

[15] Yang W, Luo P, Lin L. Clothing co-parsing by joint image segmentation and labeling[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 3182-3189.

Background Modeling: Dealing with Pan, Tilt or Zoom in Videos

Martin Radolko
University Rostock
Joachim-Jungius Str. 11
martin.radolko@igd-r.fraunhofer.de

Fahimeh Farhadifard
University Rostock
Joachim-Jungius Str. 11
Fahimeh.Farhadifard@igd-r.fraunhofer.de

Uwe Freiherr von Lukas
University Rostock
Joachim-Jungius Str. 11
uwe.freiherr.von.lukas@igd-r.fraunhofer.de

ABSTRACT

Even simple camera movements like pan, tilt or zoom constitute enormous problems for background subtraction algorithms since the modeling of the background works only under the assumption of a static camera. The problem has been mostly ignored and other algorithms have been used for videos with non-static cameras. Nonetheless, in this paper we introduce a method that adapts the background model to these camera movements by using affine transformations in combination with a similarity metric, and thereby the algorithm makes background subtraction usable for these situations. Also, to keep the generality of this approach, we first apply a detection step to avoid unnecessary adaptions in videos with a static camera because even small adaptions might otherwise deteriorate the background model over time. The method is evaluated on the extensive *changedetection.net* data set and could reliably detect camera motion in all videos as well as precisely adapt the model of the background to that motion. This does improve the quality of the background models significantly which consequently leads to a higher accuracy of the segmentations.

Keywords

Background Subtraction, Background Modeling, Video Segmentation, Change Detection, Pan, Tilt, Zoom

1 INTRODUCTION

Segmentation in videos is a particularly difficult problem of the computer vision field. Often it is the first step in a whole pipeline and all further methods are dependent on the exactness of the segmentations. The aim is to identify areas of interest in the video which can be processed further for classification, event detection and so forth. Therefore, the task for videos is usually to create a simple binary segmentation with areas of interest (foreground) and uninteresting parts (background).

The easiest scenario to create such a segmentation is a static camera because it allows the modeling of the background scene. By subtracting this model from the current frame accurate foreground-background classifications can be created in real time. For moving cameras the task becomes far more complicated and only a few algorithms have been proposed so far. Often the background modeling is skipped altogether in this scenario and instead the segmentation relies on other cues, for example the optical flow.

In this paper, we propose an addition to a background subtraction method that adapts the model to pan, tilt and zoom motions of the camera. An evaluation is done on the *changedetection.net* data set, which contains several videos with a panning or zooming camera as well as videos captured by a shaky camera. Until now, these videos have been handled with the normal algorithms for static scenes and the results were consequently unsatisfactory. Our algorithm can detect these events precisely and then lets the model mimic the motion of the camera which improved the segmentation results during camera motion without influencing the normal background subtraction for static scenes.

This research has been supported by the German Federal State of Mecklenburg-Western Pomerania and the European Social Fund under grant ESF/IV-BMB35-0006/12]

2 STATE OF THE ART

One of the most frequently used state of the art algorithms for change detection is the Mixture of Gaussian (MoG) method proposed by Stauffer et al. in [SG99]. There, the model is build of several Gaussians so that even complex scenes (swaying trees, changing lightning conditions) can be modeled accurately. Often they are used in combination with other approaches, e.g. in [WBSP14] together with a Flux Tensor. The Flux Tensor gave them a second, completely independent, cue to

their segmentation process and at the same time made the results spatially more consistent. This is important as the standard MoG is completely pixel-wise and no spatial coherency is present. A different approach for this problem was proposed in [VMZ13], there the MoG itself is enhanced with spatial method so that neighboring pixels also influence each other during the modeling process.

All of the aforementioned methods perform quite good on the standard change task for static videos but do not take into consideration the special cases of the pan, tilt and zoom videos or shaking cameras of the *changedetection.net* data set. One approach that took this into consideration is the background subtraction from [SC15], where the training images are first clustered into N groups using $K-means$ and then on each group a Single Gaussian background modeling is applied. From these different models the best is selected with a correlation coefficient, which increases the robustness to pan, tilt or zoom movement. In the overall performance the method was mediocre on the *changedetection.net* data set but on the pan, tilt and zoom videos it excelled all previous approaches.

A different approach was suggested in [FM14], where a background model based on a Neural Network is created which tries to mimic the self-organized learning behavior of the human brain. Afterwards, the model is adapted to camera movements by computing the transformation between the frame I_t and the previous frame I_{t-1} and applying it to the model.

Keypoints were used for the registration of the background model to the current frame of the video in [ACF+16]. The keypoints are matched by a K-nearest neighbor approach with the Hamming distance. However, as this method is in general prone to errors some safeguards had to be included, e.g. only pan, tilt or zoom transformations are allowed or the requirement that there are more keypoints in the background of the scene than on foreground objects.

In [ScSCJ16] motion vectors from an optical flow are used to adapt the background model to camera motion, but still only simple movements like pan or tilt can be handled easily with this method. A complete segmentation approach based on the optical flow was suggested in [OMB14]. The objects are segmented solely based on their movement vectors and therefore the method inherently can handle moving or shaking cameras quite well. The disadvantages of this approach are the slow computation time and the need of large batches of frames for an accurate calculation of the movement vectors.

3 PROPOSED APPROACH

Our approach to adapt a background subtraction algorithm for videos with pan, tilt or zoom camera movements consists of two phases. The first step is the detection of times in the video when one of the aforementioned camera movements is present. If such an event was detected the second step will compute the exact affine transformation of the background model to the current scene so that the model can be adapted correctly.

3.1 Detection of Pan, Tilt or Zoom

At first glance the detection step seems redundant, because when there is no pan, tilt or zoom of the camera the adaption step would just compute an affine transformation of zero (or very close to it) and therefore could be ignored. This is true in theory but in practice these adaptions pose a problem to the background modeling for two reasons. First, as the adaption step adapts the model of the background subtraction algorithm the comparison will always be between the current frame of the video and this model, Therefore, false detections of a panning, tilting or zooming camera can occur due to a bad background model or the presence of large foreground objects. The second problem is small misdetection e.g. due to moving objects or rounding errors at edges, which can cause small adaptions. Usually the effect of these misdetections is repaired by the algorithm itself in the next frame. However, the bigger problem of these misdetections is that every adaption of the background model is an affine transformation and causes small errors and deformations on the model which accumulate over time and deteriorate the model instead of making it better.

Hence, a good detection algorithm is necessary so that the adaption of the model only happens when it is necessary. For the detection we compare the current frame t of the video with the frame $t-2$. We chose the frame $t-2$ instead of just the previous frame $t-1$ because the constant changes like pan or tilt are more prominent then and easier to detect. This is of course very dependent on the speed of the pan or tilt as well as on the frame rate of the video, and therefore for other situations a bigger or smaller distances between the two frames which are compared might be appropriate.

The comparison between the two frames is done by applying different affine transformations on one of the frames and comparing the result with the other. The affine transformation that creates the best match is then taken as the true transformation between these two frames. To limit the possibility space of the affine transformations we confine ourselves to two basic transformations that correspond to a panning, tilting or zooming camera. First, different translations are evaluated and afterwards a transformation that corresponds to zoom is applied on top of the optimal translation.

To compare the two frames we use a similarity metric that counts the number of outliers. The reasoning behind this is that there is a natural variation between two

frames of a video, e.g. due to camera noise, but this change is usually very small. Therefore, every pixel that fulfills the following inequality

$$\|I_1(x,y) - I_2(x,y)\|_2^2 > T_{outlier}, \quad (1)$$

is counted as an outlier. In our case $T_{outlier}$ was set to 10 based on experiments. In the equation $I_1(x,y)$ is the pixel at position (x,y) of the first frame. In the end the affine transformation that produces the least outliers is taken.

With this method we obtain for every frame t a transformation that adapts it to the frame $t-2$. The extent of the transformation is then expressed in the number

$$\tau = \tau_T + w_z \cdot \tau_Z, \quad (2)$$

where τ_T is the amount of pixels that the image was translated (in x and y direction), τ_Z is the number of pixels that a corner pixel of the frame was moved by the zoom transformation and w_z is a weight parameter that controls the impact of them. As the number τ_Z is substantially smaller than τ_T for standard zoom and translation motions in videos, we gave the zoom a higher weight so that both effects have a similar influence (heuristically we choose $w_z = 5$).

The number τ is now a good measure for the camera movements we want to detect but still too sensitive to single outliers, e.g. the sudden appearances of large objects or shadows can cause erroneous calculations of τ. Therefore, we also take past calculations of τ into account to weaken the impact of single errors. This is done in the fashion of running Gaussian update by

$$\tau^{new} = (1-\alpha) \cdot \tau^{old} + \alpha \cdot (\tau_Z + w_z \cdot \tau_T). \quad (3)$$

The update rate α was set empirically to 0.1 as this allowed the detection of camera movements usually after 3 or fewer frames and is also small enough to eliminate most outliers. A smaller update rate would eliminate even more false detections but also increases the delay between the occurrence of camera movement and the detection of them by our algorithm. If this delay gets too large, an adaption of the (still untouched) background model to the already changed (over several frames) scene becomes increasingly difficult. For our data $\alpha = 0.1$ is a good compromise.

The parameter τ is very dependent on the resolution of the camera and therefore the threshold T_τ for τ should reflect this. We make the threshold depend on the *height* of the image and assume a detectable pan or tilt to have at least the size of one pixel. Since the smallest videos in the data set have a *height* of 240 pixels, this translates to the threshold $T_\tau = \frac{\text{height}}{240}$. If τ becomes larger than that a camera movement is detected and the background will get adapted accordingly (see next section). However, the threshold for τ depends also on the frame rate and the zoom or rotation speed of the camera and therefore cannot be used universally for other data sets.

3.2 Adapting the Background Model

After the successful detection of pan, tilt or zoom events the next step is the adaption of the background model. It would be easier to make this the other way around, adapt the new frame to the existing background model, because then the model can stay untouched. However, this is only possible when the camera is shaking slightly (e.g. due to wind or vibrations). When a real pan, tilt or zoom occurs this would lead, after a short time, to a situation where the incoming frame shows a completely different scene than the background model and then no adaption would be possible anymore.

To adapt the model, the first step is to extract an image from the statistical model of the background that reflects the current background so that the best affine transformation between this image and the current frame of the video can be computed. The background model in our case is created with the Gaussian Switch Model (GSM) which is a special gaussian model [RG15]. There, each background pixel is represented by two Gaussian and for every pixel and every channel we take the mean value of the currently active Gaussian and use it to create the most likely representation of the background.

Afterwards, the best affine transformation between this representation of the background and the current frame should be found. A higher accuracy (sub-pixel scale) version of the algorithm from the detection phase is used because even small errors accumulate over time and should be avoided as much as possible. The objective function used now is also different, instead of the outlier detection from Equation 1 we use

$$\sum_{x,y} \|I_1(x,y) - I_2(x,y)\|_2^2. \quad (4)$$

This function is a more exact measure of the difference between two frames and therefore better suited to determinate the precise direction and amount of movement. To lessen the impact of large foreground objects, which could disturb the accurate detection, we use the last segmentation derived from the background subtraction and exclude areas which are marked as foreground.

Similar to the first phase, we begin by looking for a translational deformation. For every direction we allow up to 10 pixels translation. To speed the process up we first look coarsely (2 pixels steps) over the whole 20 pixel range of one axis and then refined the result subsequently. This accounts already for pan, tilt or shaky

cameras but not for zoom. The zoom motion is evaluated afterwards on top of the optimal translation and the step size here is $\frac{1}{5}$ pixel. Combined, these transformations give the optimal adaption of the background model to the current frame.

This adaption now has to be applied on the complete background model, so that afterwards the segmentation of the scene and the updating of the model can succeed. For the GSM there are two Gaussian models, and the transformation has to be applied on both of them. One Gaussian model consists of two values for each pixel in the frame, mean and variance. Only jointly these values can give a comprehensive model of the background and therefore they cannot be separated by the transformation (e.g. by rounding errors) but have to represent a new pixel together after the affine transformation.

The borders pose a special problem in the adaption stage since if the background model is moved 5 pixels to the left there will be an empty space with no information on the right side of the model. After the affine transformation we identify these areas and fill them with information from the current frame of the video. The mean of the Gaussians will be set to the color value of the frame and the variance to a fixed and high value, 0.01 in our case. This ensures that the new area will be considered background in the following segmentation which is the best assumption we can make.

Lastly, to deal especially with short but fast camera motions, we evaluate the amount of foreground during times of camera motion. If there are more than 50% of the pixels classified as foreground we suppose that the model does not reflect the current scene anymore due to strong camera motion. In this case we reset the whole model and retrain it.

4 RESULTS

The method is evaluated on the comprehensive *changedetection.net* data set. It consists of 53 videos in eleven categories and two of them are *PTZ* (Pan, Tilt and Zoom) and *Camera Jitter*. In these categories there are a total of 8 videos in which the camera either is shaking or exhibits pan, tilt or zoom motions and therefore our algorithm should detect camera movements in these videos and adapt the background model accordingly. Each of the videos consists of three parts, they begin with a learning phase, then an evaluation phase for which the ground truth data is provided and lastly a part for which the ground truth data is not publically available. In this paper we only use the first two phases to assess the impact of our algorithm.

Detection Phase

In the first step we measure the detection accuracy of the proposed algorithm and afterwards the effect of the background model adaption in the *PTZ* and *Camera Jitter* videos is evaluated. The detection rate is measured by manually marking all frames with camera movement and comparing this with the results of the proposed algorithm. The results can be seen in Table 1. Since most of the videos in the data set do not exhibit camera movement there are no *True Positives* (frames in which camera movement occurred) in most categories and hence also no *false negatives*. It is vice versa in the *Camera Jitter* category since here the camera is constantly shaking. Only the *PTZ* videos contain both, times with a static camera and times with a moving camera.

The results show that the detection accuracy is very high, only in one video in the *Turbulence* category there are over 1000 false detection due to a severe heat shimmer which is difficult to differentiate from shaking. Therefore, the videos without camera movements will get hardly disturbed by the algorithm as there are only very few unnecessary background model adaptions. The detection of actual camera movements is also reliable, in the *PTZ* category the *False Negatives* are only because of the detection delay at the beginning of a movement (see equation 3). The results for the *Intermitten Pan* video are shown in detail for the first 1300 frames in Figure 1. There the detection delay at beginning can be seen as well as the false detections between the camera movements. These false detections are not a serious problem as the affine transformation applied there is usually close to zero. They would only become a problem if they would occur over longer periods without camera motion because even small transformations would then create blur effect on the model and thereby diminishes its quality.

Adaption Phase

After showing that the impact on videos with a static camera is basically nonexistent because of the few false detections, the effect of the background model adaption on the segmentations is evaluated. First the GSM background subtraction from [RG15] is used to segment the videos from the *PTZ* and *Camera Jitter* category and afterwards the same GSM algorithm is used in conjunction with the proposed model adaption. The results can be seen in Table 2 and the Figures 2 and 3 show examples from a video with a panning and zooming camera respectively.

Especially for the *PTZ* videos the results show a significant improvement over the normal background subtraction method, whereas the results for the *Camera Jitter* category show only a minor improvement overall. The reason for this is that the background modeling can deal with a shaking camera quite well, it just learns a slightly blurred version of the scene, and therefore gives still good results even without adapting the background

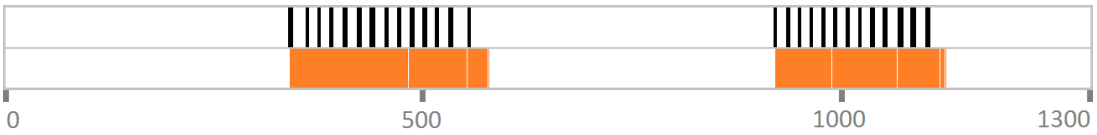


Figure 1: A detailed view on the detection accuracy on the *Intermitten Pan* video of the *PTZ* category. Shown are the results for the first 1300 frames. The top row shows the ground truth data, white areas represent times with no camera movement and black areas signify times with camera movement. The bottom row illustrates the results of our algorithm and orange areas signify detected camera motion.

model to the shaking. The adaption does improve the background model in a way that it is sharper and does include more details but this does not have a great impact on the segmentation quality.

In the pan, tilt and zoom videos the proposed algorithm shows a massive improvement because standard Background Subtraction cannot deal with these widespread camera movements. In this case the model of a normal background subtraction does not become only blurry but instead does not adapt to the movement at all (see Figure 2) or is so heavily blurred that it will not reflect the scene anymore (Figure 3). Hence, the very bad F1-Scores in this category. With our adaption algorithm the background models recreate the movements of the camera and therefore stay sharp and accurate which consequently improves the segmentation quality substantially.

The whole process of comparing different affine transformations is computational quite demanding, for a 720×480 frame the detection phase took about 0.05 seconds but if a camera movement was detected the adaption would then take another 0.8 seconds.

Figure 2: In the top row are 3 frames from the *zoomIn-zoomOut* video of the *changedetection.net* data set. The second row shows corresponding representations of the background model if the normal GSM is applied. There is almost no change in the background model although the camera zooms out because the model was trained already over a long time and adaptions now take a long time to happen. The last row shows the model with the addition of the proposed algorithm where the model adapts to the zooming motion.

5 CONCLUSION

In this paper an approach for the adaption of background models to pan, tilt or zoom camera movements is proposed. The method consists of two steps and the first part is the detection of camera movement. This is important to avoid unnecessary and potentially inaccurate adaptions of the model. We have shown on the large *changedetection.net* data set that the proposed detection works very accurately and therefore the whole algorithm does barely affect the background subtraction on videos with a static camera. If camera movement is present, the second phase adapts the background model to the slightly changed scene due to the moving camera movement and thereby we can improve the background model consistently. These sharper and more accurate models lead to overall significantly increased segmentation qualities when camera movement is present.

6 REFERENCES

[ACF+16] Danilo Avola, Luigi Cinque, Gian Luca Foresti, Cristiano Massaroni, and Daniele Pannone. A keypoint-based method for background modeling and foreground detection using a {PTZ} camera. *Pattern Recognition Letters*, 2016.

[FM14] A. Ferone and L. Maddalena. Neural background subtraction for pan-tilt-zoom cameras. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):571–579, May 2014.

[OMB14] P. Ochs, J. Malik, and T. Brox. Segmentation of moving objects by long term video analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1187–1200, June 2014.

[RG15] Martin Radolko and Enrico Gutzeit. Video segmentation via a gaussian switch background-model and higher order markov random fields. In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications Volume 1*, pages 537–544, 2015.

[SC15] H. Sajid and S. C. S. Cheung. Background subtraction for static amp; moving camera. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4530–4534, Sept 2015.

[ScSCJ16] Hasan Sajid, Sen ching S. Cheung, and Nathan

Category	Bad Weather	Baseline	Camera Jitter	Dynamic Background	Intermittent Object	Shadow	Thermal	Low Framerate	Night Videos	PTZ	Turbulence	Overall
TN	11950	6050	0	18874	18629	16950	21100	6503	10466	3270	8034	121826
TP	0	0	5204	0	0	0	0	0	0	1641	0	6845
FP	0	0	1216	0	0	0	0	0	0	221	0	1437
FN	0	0	0	0	21	0	0	297	0	738	1566	2622

Table 1: Results of the camera movement detection of the proposed algorithm. The *changedetection.net* was used for the evaluation and shown are the numbers of frames that were correctly or falsely classified. For example *True Negatives* are the frames in which no camera movement was present and which were correctly classified as such.

Video	F1-Score		Video	F1-Score	
Camera Jitter	only GSM	proposed approach	PTZ	only GSM	proposed approach
Badminton	0.8476	0.8733	continuousPan	0.0909	0.5501
Boulevard	0.5983	0.6266	intermittenPan	0.0768	0.5744
Sidewalk	0.5423	0.5228	twoPositionPTZCam	0.1988	0.7327
Traffic	0.7288	0.7528	zoomInzoomOut	0.0221	0.3781

Table 2: Compared are the results of the GSM background subtraction with and without the addition of proposed algorithm. A clear improvement can be seen, especially for the *PTZ* videos.

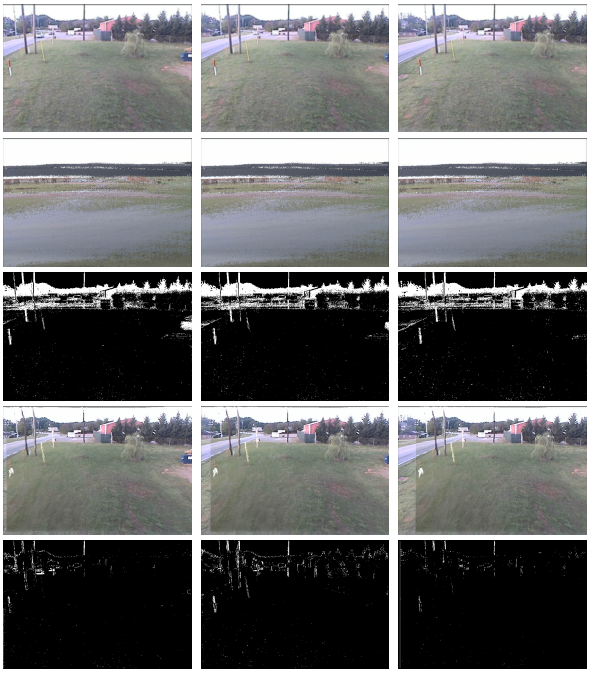


Figure 3: Similar to Figure 2 are here shown the results of the *continuousPan* video. The background model created with the normal background subtraction algorithm in the second row is extremely blurry because of the constant panning of the camera. Hence, the segmentation results show a lot of false detections. The proposed adaption to this pan makes the background model sharper and more accurate so that most false detection vanish (row four and five).

Jacobs. Appearance based background subtraction for {PTZ} cameras. *Signal Processing: Image Communication*, 47:417 – 425, 2016.

[SG99] Chris Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Vol. Two*, pages 246–252, 1999.

[VMZ13] S. Varadarajan, P. Miller, and H. Zhou. Spatial mixture of gaussians for dynamic background modelling. In *2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 63–68, Aug 2013.

[WBSP14] R. Wang, F. Bunyak, G. Seetharaman, and K. Palaniappan. Static and moving object detection using flux tensor with split gaussian models. In *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 420–424, June 2014.

Efficient Atmospheric Rendering For Mobile Virtual Globes

José M. Santana
University of Las Palmas de G.C. Imaging Technology Center (CTIM) Spain (35017), Las Palmas de G.C.
josemiguel.santana @ulpgc.es

Agustín Trujillo
University of Las Palmas de G.C. Imaging Technology Center (CTIM) Spain (35017), Las Palmas de G.C.
agustin.trujillo @ulpgc.es

José P. Suárez
University of Las Palmas de G.C. Division of Mathematics, Graphics and Computation (MAGiC), IUMA, Spain (35017), Las Palmas de G.C.
josepablo.suarez @ulpgc.es

Sebastián Ortega
University of Las Palmas de G.C. Division of Mathematics, Graphics and Computation (MAGiC), IUMA, Spain (35017), Las Palmas de G.C.
sebastian.ortegatrujillo @gmail.com

ABSTRACT

Most virtual globe systems feature a rendering of the atmosphere that surrounds the earth. This element is so widespread that seeing a virtual earth without a surrounding halo makes the image seem much more artificial. Atmospheric rendering is a costly process aimed to enhance the realism and beauty of the scene. For that reason many virtual globe systems, specially in low-resourced devices, rely on simplified schemes to represent the atmosphere. However, the accurate representation of the atmosphere implies the volumetric rendering of the semi-transparent air mass that covers the whole geographical scene. Thus, the color of each pixel is composed of the light scattered by all the points in space projected on that pixel. The present work takes advantage of the spherical symmetry of the atmosphere's mathematical model to implement efficiently this volumetric rendering on mobile devices.

Keywords

Virtual Globe, Atmospheric Rendering, Volumetric Rendering, Rayleigh Scattering, Mobile Devices

1 INTRODUCTION

Volumetric rendering on mobile devices is a challenging field [9] and there are many proposals for general purposed volume visualizations, most of them based on 2D/3D textures that describe the composition of the gas. Fortunately, the features of the atmospheric model allows us to tackle it in an analytical manner. The contribution presented in this work consists of an implementational study of atmospheric *Rayleigh Scattering* that uses state of the art techniques. The atmosphere presents other visual effects like *Mie Scattering*, which represents the scattering due to particles of a comparable size to the wavelengths of visible light, for instance water droplets in clouds. However, these other effects are out of the scope of this research, and some of them use techniques very similar to the ones here presented. Hence, the final goal is to obtain a realistic rendering of the *Rayleigh scattering* effect using the tools and hardware limitations of an OpenGL ES 2.0 supported mobile device. To that effect, the 3D problem is translated to several 2D domains in which the problem is computationally reduced.

2 ATMOSPHERIC MODEL

The atmosphere of the earth can be thought of as a spherically-symmetric envelope of pure air. Considering also a spherical model of the earth, centred at the axis origin, it extends from the planetary surface (at a radius R_E of $6,371 \times 10^9$ m.) to an undefined height, decreasing in density as we move upwards.

The density distribution of such a mass of gas can be fairly represented in terms of an exponential formula, $D(h) = D_{max} * exp(-h/H_0)$ where $p(h)$ is a density factor and H_0 can be assigned 7994, following the work of Nishita and Sirai [7].

However, in this work we are only considering the illumination contributions of the troposphere and the stratosphere, which reaches a height H_{Max} of 50 km. Color contributions of upper layers are negligible and can be dismissed.

The real atmosphere thickness is around $8,3 \times 10^{-4}$ % of the radius of the earth. Hence, our final implementation includes a scale factor for visualization purposes, as other systems apply to the rendering of terrain orography. The radius of the final atmosphere, R_A from now on, is of $50 \times 10^3 \times S + R_E$ m., being S the scale factor. Likewise, the density factor must be rescaled as $H_0' = H_0 \times S$.

On our model the sun is placed in the position P_S at a distance 149.6×10^9 m. from the center of the scene on the X axis. All incoming light can be considered then

parallel, pointing on the X positive direction. Hence, a cylindrical shadow is projected in the same direction which is taken into account on our calculations.

As previously mentioned, the two most notable forms of atmospheric scattering are Rayleigh scattering and Mie scattering. Rayleigh scattering is caused by small molecules in the air, and it scatters light more heavily at shorter wavelengths (blue first, then green, and then red). In our rendering model, the atmosphere is composed by an isotropic pure air, so the only modeled interaction is Rayleigh scattering, which is responsible for most atmospheric effects, the predominant blue tone of the daytime sky, as well as the reddish dawns and sunsets.

There have been several attempts to accurately render the atmosphere in VG systems [8, 6]. The work of [7] has been specially relevant due to its clear understanding of the underlying numerical analysis of the light spectrum. A bright example of a GPU implementation is given by Sean O'Neil in the book [10]. Other papers have addressed more advanced effects, such as smoke and cloud rendering [4], the light reflected by different parts of the earth or mountain shadows (light shafts) [1].

3 THE ATMOSPHERIC RAYLEIGH SCATTERING MODEL

The model considers the refracted light in the direction of the observer given the illumination of the view ray by a directional light source (the Sun). Based on the Rayleigh scattering equation, the amount of sunlight of a particular wavelength refracted by a particular point in the direction of the camera is given by the equation:

$$F(\theta,g) = \frac{3(1-g^2)}{2(2+g^2)} \times \frac{(1+cos^2(\theta))}{(1+g^2-2g\times cos(\theta))^{3/2}} \quad (1)$$

$$I(\lambda,\theta,h) = \frac{I_0(\lambda)KFr(\theta)}{\lambda^4} * \rho(h) \quad (2)$$

This equation is based on the following terms:

- $I_0(\lambda)$ is the incoming intensity of sunlight for a given wavelength λ (in nanometers).
- $Fr(\theta)$ is the phase function, in this case implemented as the Henyey-Greenstein function, with a $g = 0$.
- K is a constant based on the molecular density of the atmosphere (see [7]).
- $\rho(h)$ is the density coefficient explained in Section 2.

To compute the final light incident on our camera, it is also necessary to take into account the amount of light scattered out by upper layers of the atmosphere before reaching the ray P_aP_b. Besides, the light is scattered out in its way to the observer by the air in the ray P_aP_b itself.

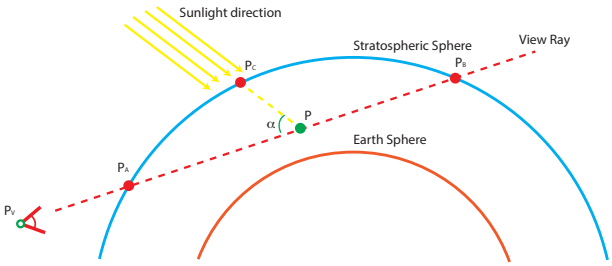


Figure 1: Geometrical model of a view ray through the atmosphere.

Both terms are based on the *Optical Depth* of the rays P_cP and PP_V respectively. P represents each of the points of the main ray, P_C is the point where the sunlight hitting P enters the atmosphere and P_V is the camera position. The physical system is depicted in Figure 1: By applying the *Beer's Exponential Extinction Law*, we can determine that light will be scattered out exponentially to those optical depths, resulting in:

$$t(P_aP_b,\lambda) = \frac{4\pi K}{\lambda^4} \int_{S(P_a)}^{S(P_b)} \rho(h(P(s)))ds \quad (3)$$

$$W(\lambda,\theta) = I_0(\lambda)F_r(\theta)\frac{K}{\lambda^4} \quad (4)$$

$$O(P_a,P,P_c) = exp(-t(P_aP,\lambda) - t(P_cP,\lambda)) \quad (5)$$

$$I(\lambda) = W(\lambda,\theta) \int_{S(P_a)}^{S(P_b)} \rho(h(P(s)) \times O(P_a,P(s),P_c(s))ds \quad (6)$$

4 FULL-SPECTRAL RENDERING

The light emitted by the sun can be fairly modeled as the radiation of a black-body, which is an opaque object at thermodynamic equilibrium. This radiation is classically described by Planck's equation. Considering the standardized color space CIE1931, the visible spectrum of light ranges from 380 nm. to 780 nm. of wavelength.

To convert these wavelengths to specific colors in the XYZ space of CIE 1931, the *Simple, Single-Lobe Fit* model has been used. Assuming a sRGB display, an approximation of these values can be achieved by a linear transformation. The tristimulus components RGB of such colors are precalculated, so they do not need to be computed on each frame.

5 EXPLOITING MODEL SYMMETRY

Due to the spherical symmetry of the atmospheric density distribution, we can define a coordinate system so ray $\overrightarrow{AB}$ is in the XY plane and parallel to Y.

$$M = \begin{bmatrix} \hat{D} \\ \hat{C} \\ \hat{C}\times\hat{D} \end{bmatrix} \text{ where } \begin{matrix} D = (B \hat{-} A) \\ C = A + (-A\cdot D)D \end{matrix} \quad (7)$$

The starting and ending points of the ray can therefore be computed as $P_A = M \times A'$ and $P_B = M \times B'$. The points in this 2D space can be expressed in polar coordinates, being $h = \sqrt{x^2 + Y_0^2}$, and $\varphi = atan2(Y_0, x)$. Note that Y_0^2 can be precomputed for all points along the ray. This new parametrization of a ray traversing the atmosphere can be seen in Figure 2.

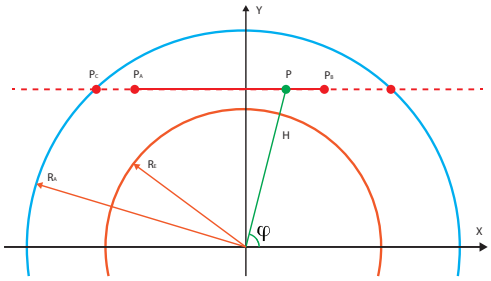


Figure 2: Ray system simplified to 2D equivalent.

6 PRECALCULATING OUT-SCATTERING OPTICAL DEPTHS

The current proposal offers some computational shortcuts to obtain the out-scattering coefficients needed to compute the full atmospheric model. The main idea was first proposed in the work of Nishita and Sirai [7] and to a further extent by Woolley [10]. However, they do not explicitly apply the transformation explained in Section 5 from which these caching techniques benefit, nor explore the limits of our integrating function due to inherent symmetries.

The out-scattering factors require the integration of the rays $\overrightarrow{P_C P}$ (coming from the sun) and $\overrightarrow{P_A P}$ (coming from the observer). By translating these two rays to our 2D plane, these two integrals can be expressed in the form:

$$f(X_0, X_1, Y_0) = \int_{X_0}^{X_1} \rho(h(x, Y_0))dx \quad (8)$$

$$f(X_0, X_1, Y_0) = g(X_1, Y_0) - g(X_0, Y_0) \quad (9)$$

where $g(X,Y) = \int_{-R_A}^{X_1} \rho(h(x, Y_0))dx$.

This function can be transformed to polar coordinates as $g(\varphi, h)$ offering a much more compact, rectangular definition of its domain, as seen in Figure 3. This table can be further compressed, taking into account the following rotational symmetries:

- $g(\varphi, h) = g(-\varphi, h)$.
- For $h > R_A$, $g(\varphi, h) = 0$.
- For $h < R_E$, $g(\varphi, h) = \infty$.
- For $0 \leq \varphi \leq \pi/2$, $g(\varphi, h) = g(\pi/2, h * \sin(\varphi)) + (g(\pi/2, h * \sin(\varphi)) - g(\pi - \varphi, h)) = 2 * g(\pi/2, h * \sin(\varphi)) - g(\pi - \varphi, h)$.

Thus, $g(\varphi, h)$ is fully represented by mapping the domain $\varphi = \pi/2 - \pi$ and $h = R_E - R_A$. As a counterpart, a reading of the optical depth function, $f(X_0, X_1, Y_0)$ could require two accesses to the table, or used texture. It is also important to note that, due to the polar nature of the function $g(\varphi, h)$ we lose spatial resolution as we move upwards, as it is pointed out by [1]. Ensuring a minimum spatial resolution of n meters requires an angular resolution of $\Delta\alpha = \arcsin(\frac{n}{E_A})$.

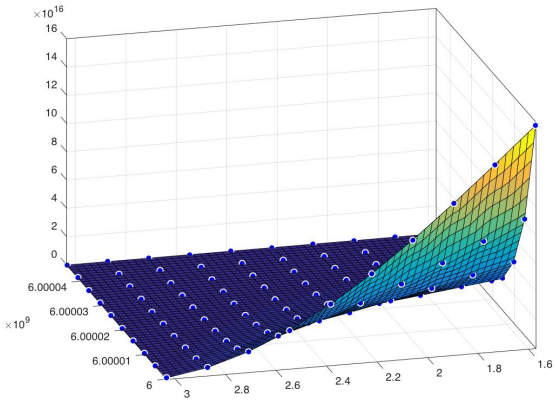


Figure 3: Optical depth map in polar coordinates.

7 COMPUTING INTERPOLATION POINTS

As a contribution, the present work proposes the use of a precomputed array of an odd length (n) of the form $V_i = 1 - exp(-\frac{\mu}{n} \times i)$. For rays starting outside of the atmosphere, the highest points are separated from the outer atmosphere a ratio of $exp(-\mu)$, thus a value of $\mu = 2.5$ produces a separation of 0.08 which seems reasonable. Using the same formula, two other vectors V' and V'', of lengths $\lceil \frac{n}{2} \rceil$ and $\lfloor \frac{n}{2} \rfloor$ respectively, are precalculated. Once the ray has been transformed to the X_0, X_1, Y_0, form it is determined whether it crosses the Y axis ($X_1 > 0$) or not.

For rays not crossing that line, intersection points P_i are calculated by Expression 10, and rays that do by Expression 11.

$$P_i = \{X_0 + V_i * (X_1 - X_0), \quad Y_0\} \quad (10)$$

$$P_i = \begin{cases} \{X_0 - V'_i * X_0, & Y_0\} & \text{if} \quad i < n/2 \\ \{X_0 + V''_i * X_1, & Y_0\} & \text{if} \quad i > n/2 \end{cases} \quad (11)$$

8 NUMERICAL ANALYSIS OF THE OPTICAL DEPTH MAP

The shading algorithm has been fully implemented in a Matlab program that allows to render the atmosphere from an outside perspective. This allowed us to perform numerical analysis on the intermediate results of

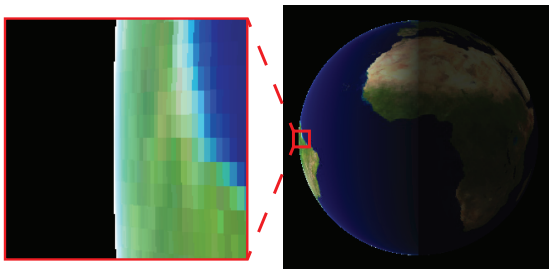


Figure 4: Final atmospheric rendering (full frame and close-up) at real scale.

the process as numerical images. Figure 4 presents a test image for not scaled atmosphere.

8.1 Simplification of the optical depth map

The out-scattering intensity 2D map proposed in Section 6 avoids two optical depth integrations of the final scattering model. The most straight-forward way to access such map for the shading program is to have the map as a preallocated 2D texture. However, this map is accessed up to four times for rays that cross the atmosphere, implying 4 accesses to GPU memory. Besides the read texture must be interpolated which internally derives in further memory access. In order to speed this process up, a numerical fit of the $g(\varphi,h)$ grid data is proposed. However, after several attempts, it seems difficult to adjust properly a single 3D function to such data. Instead, a multi-curve adjustment at linearly spaced heights is proposed. Our best approximation uses 10 different equations achieving a mean error of 2.84%. All these equations can be coded into the GLSL program, so any value $g(\varphi,h)$ can be interpolated from two polynomial evaluations.

8.2 Effects of view ray sampling

The light scattered in our direction by each ray is computed as a function of the amount of air it traverses and the amount of incoming lights. These punctual contributions have to be integrated along the ray at a number of locations that is implementation-specific. As an approach to reduce the computational costs of the rendering, we intend to minimize the number of sampling points used to estimate the contribution of a particular view ray. To that end, a numerical study of the different wavelength intensities has been conducted.

As depicted in Figure 5, the relative error achieved in the final image decreases as we measure the contributions of more intermediate points. However, the addition of more sampling points seems to reduce the error following a logarithmic curve. Thus, according to experimentation reducing from 13 to 5 sampling points only introduces a $16,15\%$ of error, while improving the performance by a $2,6$ speed-up factor.

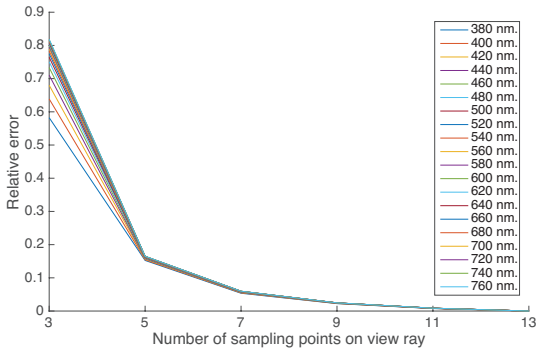


Figure 5: Wavelength intensity error vs. number of sampling points.

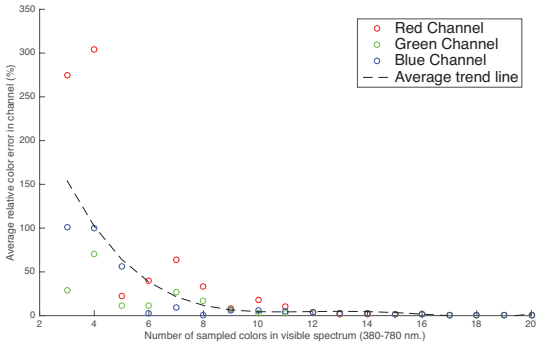


Figure 6: Color relative error due to spectrum sampling.

8.3 Effects of visible light spectrum sampling

As there is an uncountable infinity of possible wavelengths of light on the sun radiation, the spectrum must be sampled on several points and their color contributions integrated. This final experiment intends to show the effects of the sampling density of the visible spectrum of light wavelengths. As previous examples, we find the approximation of [5] that uses 14 integration points. The goal is to minimize the number of sampled wavelengths to produce an accurate rendering. To that end, the visible spectrum has been split in intervals of similar size that are represented by their central value. The number of samples in this experiment ranges from 3 to 20, being the "ground truth" the color achieved by this last partitioning.

This numerical study shows that the error tends to stabilize rapidly, as can be seen in Figure 6. The final sampling is discretionary and case-dependant but 10 samples offer a 9.81% error and hence seems to be a reasonable choice for many applications.

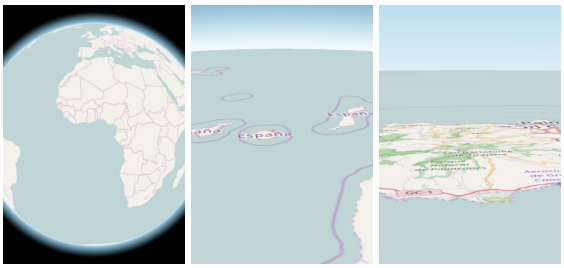
9 CONCLUSIONS AND FUTURE WORK

The here-presented ongoing research proposes some strategies aimed to reduce the number of operations

needed to render atmospheric effects on virtual globe environments.

To that end, the mathematics behind the Rayleigh scattering and the model of the atmosphere's gas distribution are analyzed. From this analysis, a simplified 2D representation of the involved light rays has been achieved. Thus, the overall number of needed calculations is reduced. Moreover, a compact optical depth map is generated.

This 2D simplification of the rendering algorithm has been successfully implemented on a simplified version of the atmospheric model. This first approximation disregards the out-scattering factors but at this point offers a semi-realistic look, running at 60 FPS on several tested iOS and Android devices.



(a) Outer view. (b) Inner view. (c) Ground view.

Figure 7: Atmospheric rendering. Atmosphere height exaggerated x15. This implementation was carried out using the Glob3 Mobile engine [11].

Finally, it has been experimentally demonstrated the effects of reducing the sampling rate both geometrically on the projected rays and on the light spectrum. The conducted numerical simulations suggest that 10 sampled wavelengths and 5 intermediate points per ray may be sufficient.

As future work, we intend to implement the full model on both desktop and hand-held device in order to confirm its applicability on different GPU architectures. On that basis, new factors could be added to the model, such as a Mie Scattering component, in order to enhance the realism of the atmospheric rendering.

10 ACKNOWLEDGMENTS

The first author wants to thank Agencia Canaria de Investigación, Innovación y Sociedad de la Información, and the European Social Fund, for the grant "Formación del Personal Investigador-2012" that made possible this work.

The fourth author wants to thank Universidad de Las Palmas de Gran Canaria for its grant "Programa de personal investigador predoctoral en formación 2015", which made possible this work.

11 REFERENCES

[1] Eric Bruneton and Fabrice Neyret. Precomputed atmospheric scattering. *Computer Graphics Forum*, 27(4):1079–1086, 2008.

[2] Richard L. Burden and John Douglas Faires. *Numerical Analysis*. 2011.

[3] Yoshinori Dobashi. Interactive Rendering of Atmospheric Scattering Effects Using Graphics Hardware. *Hwws*, pages 99–109, 2002.

[4] Yoshinori Dobashi, Kazufumi Kaneda, Hideo Yamashita, Tsuyoshi Okita, and Tomoyuki Nishita. A simple, efficient method for realistic animation of clouds. *Siggraph*, pages 19–28, 2000.

[5] John Irwin. Full-Spectral Rendering of the Earth's Atmosphere using a Physical Model of Rayleigh Scattering. *Eurographics*, 1996.

[6] Henrik Wann Jensen, Frédo Durand, Julie Dorsey, Michael M Stark, Peter Shirley, and Simon Premože. A physically-based night sky model. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 399–408. ACM, 2001.

[7] Tomoyuki Nishita and Takao Sirai. Display of The Earth Taking into Account Atmospheric Scattering. *Siggraph*, (2):175–182, 1993.

[8] Tomoyuki Nishita, Yasuhiro Miyawaki, and Eihachiro Nakamae. A shading model for atmospheric scattering considering luminous intensity distribution of light sources. *ACM SIGGRAPH Computer Graphics*, 21(4):303–310, 1987.

[9] Jose M Noguera and J Roberto Jimenez. Mobile Volume Rendering: Past, Present and Future. *Ieee Transactions on Visualization and Computer Graphics*, 22(2):1164–1178, 2016.

[10] Cliff (University of Virginia) Woolley. *GPU Gems 2*, volume 2. 2005.

[11] José Miguel Santana, Jochen Wendel, Agustín Trujillo, José Pablo, Alexander Simons, and Andreas Koch. Multimodal Location Based Services - Semantic 3D city data as Virtual and Augmented Reality. In G Gartner and H Huang, editors, *Progress in Location-Based Services 2016*, chapter 17. Springer, 2016. ISBN 978-3-319-47289-8. doi: 10.1007/978-3-319-47289-8{_}17.

An Automatic Hole Filling Method of Point Cloud for 3D Scanning

Yuta MURAKI
Osaka Institute of Technology
Osaka, Japan
yuta.muraki@oit.ac.jp

Koji NISHIO
Osaka Institute of Technology
Osaka, Japan
koji.a.nishio@oit.ac.jp

Takayuki KANAYA
Hiroshima International University
Hiroshima, Japan
t-kanaya@hw.hirokoku-u.ac.jp

Ken-ichi KOBORI
Osaka Institute of Technology
Osaka, Japan
kenichi.kobori@oit.ac.jp

ABSTRACT

In recent years, due to the development of three-dimensional scanning technology, the opportunities for real objects to be three-dimensionally measured, taken into the PC as point cloud data, and used for various contents are increasing. However, the point cloud data obtained by three-dimensional scanning has many problems such as data loss due to occlusion or the material of the object to be measured, and occurrence of noise. Therefore, it is necessary to edit the point cloud data obtained by scanning. Particularly, since the point cloud data obtained by scanning contains many data missing, it takes much time to fill holes. Therefore, we propose a method to automatically filling hole obtained by three-dimensional scanning. In our method, a surface is generated from a point in the vicinity of a hole, and a hole region is filled by generating a point sequence on the surface. This method is suitable for processing to fill a large number of holes because point sequence interpolation can be performed automatically for hole regions without requiring user input.

Keywords

Hole filling, 3D Scanning, Surface Approximation, Point Cloud

1. INTRODUCTION

In recent years, due to the development of three dimensional scanning technology, the opportunities for real objects to be three-dimensionally measured, taken into the PC as point cloud data, and used for various contents are increasing. In three-dimensional scanning, data of the entire surroundings of the scanning object is acquired by generally measuring the scanning object from a plurality of directions and aligning the obtained plural point group data. In Figure 1, the point cloud data obtained from a plurality of directions are displayed in a color-coded manner, and one model data is constructed by integrating them.

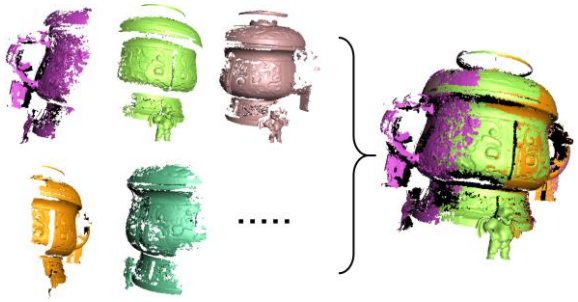
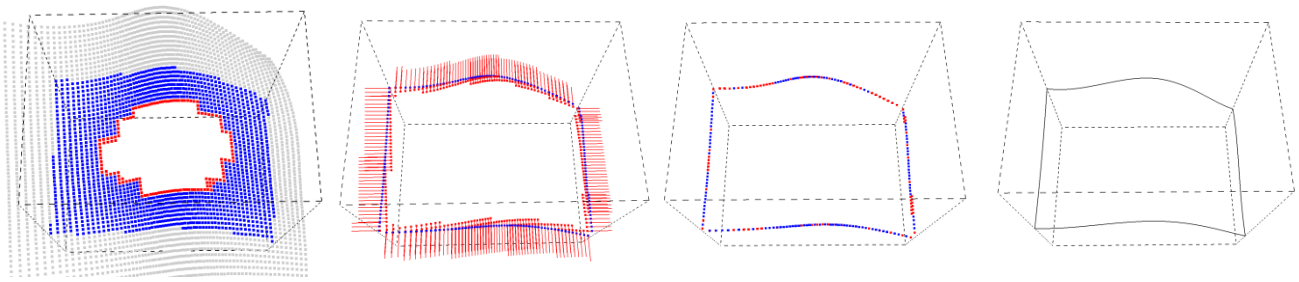


Figure 1. The point cloud data obtained from three-dimensional measurement.

However, the point cloud data obtained by three-dimensional scanning has many problems such as data loss due to occlusion and the material of the object to be measured, and occurrence of noise. Therefore, it is necessary to edit point cloud data obtained by scanning. Especially, since the point cloud data obtained by scanning contains many data missing, it takes a huge amount of time to fill holes.



(a) Selection of point group **(b) Projection to OBB plane** **(c) Extracting sample points** **(d) Generate closed region**

Figure 2. Procedure for generating a closed area.

Therefore, we propose a method to automatically fill hole of point cloud data obtained by three-dimensional scanning. In this method, the bounding box (OBB) is generated from the neighboring point of the missing region and the region is expanded. After that, a point group included in the expanded bounding box is extracted and approximated to generate a B-spline surface. Then, a point sequence is generated on the generated B-spline surface and interpolation is performed by deleting points other than the portion corresponding to the missing. This method is suitable for processing to fill a large number of holes because point group interpolation can be performed automatically for hole regions without requiring user input.

2. RELATED WORK

The hole filling method of point cloud data is roughly divided into the following three types.

1. Method of interpolating using a database[1,2].
2. Method of interpolating using the shape information of the entire model[3,4].
3. Method of interpolating using the surrounding shape[5-11].

The method [1, 2] of interpolating using a database is a method of selecting a shape similar to the target object from the database and integrating it with the target object. In these methods, there is a problem that it is costly because a shape similar to the target object needs to be modeled and stored in advance.

As a method of using the shape information of the entire model, a method of copying an area having the maximum degree of similarity of shape and texture has been proposed [3, 4]. However, in method [3], hole is filled with one copy. Therefore, when there is no similar shape of the same size as the hole in the model, it is an interpolation result which is liable to cause discomfort.

In the method [4], interpolation of the hole is performed by copying a shape with the minimum energy function by using similarity of shape and texture. In this method, there are problems that the parameters are determined manually and that the interpolation result depends greatly on the parameters. In addition, there is a problem that these methods [3, 4] commonly have high calculation cost.

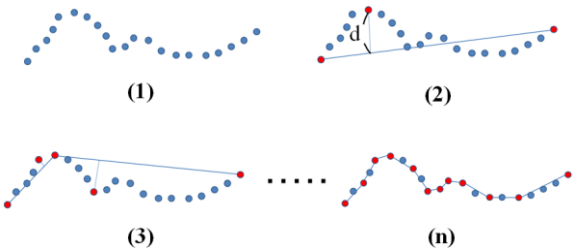
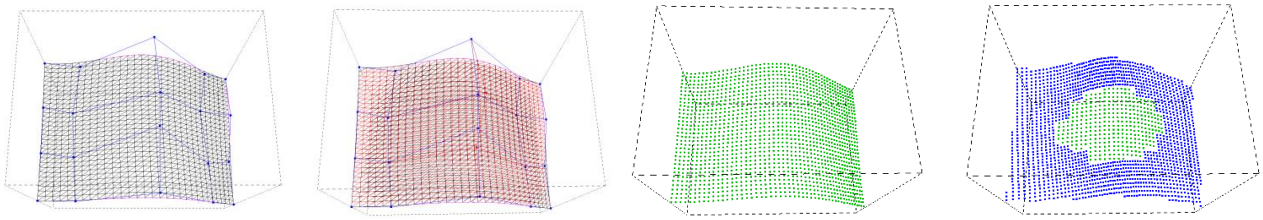


Figure 3. Piecewise linear approximation.

The method [5-11] that uses surrounding shapes for interpolation is a method of performing smooth interpolation by using differential equations and surface fitting. In these methods, good results can be obtained for small hole regions. In particular, the method [9] is also applicable to complex hole regions with "islands". However, when large hole regions are filled, there are cases where shapes greatly different in the properties of three-dimensional structures are generated. If the area of the hole is larger, these method produce plane based results.

3. PROPOSED METHOD

We propose a hole filling method based on surface approximation. In our method, by approximating neighboring points of a hole region, a surface covering a hole region is generated and an interpolation point is generated. In the proposed method, the control point is obtained so that the error between the surface to be generated and the point representing the outline of the hole is small, so it is possible to calculate an interpolation point with relatively high accuracy. In addition, since the continuity with the point group around the hole is taken into consideration, the interpolation points are small in gap with input point cloud.



(a) Calculation of internal control point **(b) Refine the surface control point** **(c) On-surface point generation** **(d) Interpolation point generation**

Figure 4. Curved surface and interpolation point generation

3.1. Selection of point group

In this section, a method of extracting a point group used for surface approximation will be described with using a part of the input point cloud shown in Figure 2 (a).

First, as shown by the red dots in the Figure 2 (a), points near the missing region representing the outline of the hole are extracted from the input point cloud by using based on Gerhard's method [12]. Then, a bounding box (OBB) of the extracted points is generated. Then using the vector obtained by PCA (principal component analysis), the area is expanded by expanding the generated OBB to the outside. In this method, as shown by blue dots in the Figure 2 (a), a point group included in the expanded OBB is extracted from the input point cloud as a point group to be used for fitting a surface.

3.2. Surface fitting

In this section, we describe a method to automatically generate a surface by using point group extracted in Section 3.1.

3.2.1. Generation of boundary curve

We describe the generation of a boundary curve of a fitted B-spline surface. First, as shown by the red dots in Figure 2 (b), neighbor points of each plane constituting the expanded OBB are extracted from the point group obtained in Section 3.1. Next, as shown by the red line in the Figure 2 (b), the extracted neighboring point of the OBB plane is projected onto the OBB plane to generate a new point sequence indicated in blue. After that, a sequence of points is extracted based on piecewise linear approximation from a number of blue points on the OBB plane. As shown in Figure 3, piecewise linear approximation is a method of approximating a point sequence with line segments and repeating recursive division until the distance d of the point farthest from the line segment is within the threshold ε. As a result, characteristic points can be extracted. The extraction result is indicated by red dots in Figure 2 (c). The point sequence obtained as described above is approximated by a B-spline curve.

The four generated B-spline curves are independent each other, and the end points are not connected. Therefore, four neighboring B-spline curves are connected by extracting the neighboring points of the line segment representing the four corners of the OBB one by one and adding it to the control point of the B-spline curve. Then, as shown in Figure 2 (d), a closed quadrilateral region is generated.

3.2.2. Calculate internal control points of surface

As shown in Figure 4 (a), using the B-spline curve generated in Section 3.2.1 and the point group obtained in Section 3.1, the internal control points of the B-spline surface are calculated by the least squares method. The generated control points of B-spline surface is indicated by blue in Figure 4 (a). The generated polygons of B-spline surface is indicated by black in Figure 4 (a).

3.2.3. Refinement of internal control points

Since the curved surface generated by this method is an approximation of a point group, the center part of the surface region where the hole exists generally has low accuracy. Therefore, the accuracy of the interpolation point is improved by recalculating the internal control points so that the error of the hole neighborhood point becomes small. Polygons of B-spline surface and the surface control points after the refinement processing are indicated by red in Figure 4 (b).

3.3. Generate interpolation points

As shown in Figure 4 (c), a point sequence is generated on the generated B-spline surface. In order to eliminate discomfort to the input point cloud, the interval of the generated point sequence is calculated by the ratio of the length in the u and v direction of the generated quadrilateral region. After that, as shown by green dots in Figure 4 (d), distance from the generated surface points are extracted and set as interpolation points. This method is not influenced by the outline shape of the missing region.

Table 1. Accuracy evaluation

Data Set	#Original points	#Vicinity points	#Fill points	#New points	#control points	error (%)	time (sec.)
Data A	4,977	522	168	5,145	16	0.482436	0.19
Data B	4,770	600	155	4,925	16	0.344566	0.29
Data C	4,370	746	154	4,524	16	0.335054	0.24
Data D	9,419	647	190	9,609	16	0.285562	0.24
Data E	2,034	795	219	2,253	20	0.813469	0.27
Data F	4,375	1,636	366	4,741	20	0.302318	0.55
Data G	8,854	1,901	603	9,457	16	0.496557	0.69
Data H	8,349	1,620	714	9,063	16	0.532111	0.58
Data I	22,093	5,607	1,888	23,981	20	0.524364	3.27

In other words, it can also be applied to complex holes such as including "island" [9] where points exist inside the hole.

4. EXPERIMENTAL RESULTS

In order to verify the effectiveness of this method, accuracy and processing time were measured by applying this method by generating holes for randomly generated surface shapes. After that, we applied this method to actual scan data and verified the usefulness of the method. For the experiment, we used a desktop PC with *Core i7 (3.4 GHz)* and *8GB*.

4.1. Application to experimental data

Figure 5 shows the results of applying our method to nine point cloud data including hole. Gray points are point cloud with hole, and green points are interpolation points. Also, the generated B-spline surface is indicated by black polygons.

4.2. Accuracy evaluation

In this section, shape evaluation of the generated interpolation points will be described. In order to verify the accuracy, the distance between the generated interpolation points and the true value (surface) was measured. Table 1 shows the results of shape evaluation. Vicinity points in Table 1 are the points used for surface generation. Fill points are interpolation points generated by our method. Error indicates the accuracy of the interpolation points and is the value obtained the average of the distance between the interpolation points generated by our method and the true value (surface), and normalizing by the bounding box size. As shown in Table 1, it was confirmed that the error of the interpolation points is less than 1%, and approximation can be performed with good precision. In addition, the processing time is less than 1 second, and practicality is also proved.

4.3. Application to actual scanning data

Figure 6 shows the results of applying our method to scanning data obtained from a 3D scanner. Figures 6 (a), (c) and (e) are input data including hole, and (b), (d) and (f) are the result of applying the method. We can see that interpolation that maintains the shape features is possible.

5. CONCLUSIONS

In this paper, we proposed a hole filling method for point cloud based on surface approximation. In this method, a point group to be used for approximation is extracted by enlarging a region from a hole neighbor point, and it is possible to automatically generate an interpolation point. Moreover, by using refined outline information and performing refinement processing so that the accuracy of the interpolation point is high, high precision hole filling was made possible. We applied our method to the hole region including concavities and convexes and confirmed that high precision and high speed interpolation is possible. In addition, applying our method to actual scanning data, we confirmed that it is possible to fill hole without feeling uncomfortable.

It is difficult to apply our method to the part where the point near the hole becomes irregular like the base of the handle. In addition, if the area where the points are missing is none smooth then our method may not produce satisfactory results. Therefore, adaptation to more shapes can be cited as future works. Further improvement of the accuracy of the interpolation point can be cited as future works.

6. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number JP 17K13571.

7. REFERENCES

[1] V. Kraevoy and A. Sheffer, "Template-based mesh completion", Proc. Eurographics Symp. On Geometry Processing, pp.13-22, 2005.

[2] M.Pauly, "Example-based 3D scan completion", Proc. Eurographics Symp. On Geometry Processing, pp.23-32, 2005.

[3] S. Park, X. Guo, H. Shin, H. Qin, "Shape and Appearance Repair for Incomplete Point Surfaces", Proc. ICCV, Vol. 2, pp.1260-1267, 2005.

[4] N. Kawai, A. Zakhor, T. Sato, N. Yokoya, "Surface Completion of Shape and Texture Based on Energy Minimization", Proc. ICIP, pp.913-916, 2011.

[5] J. Verdera, V. Caselles, M. Bertalmio and G. Sapiro, "Inpainting surface holes", Proc. Int. Conf. on Image Processing, Vol.2, pp.903-906, 2003.

[6] P. Liepa, "Filling Holes in Meshes. Symposium on Geometry Processing", pp.200-205, 2003.

[7] J. Wang, M. M. Oliveira, "Filling holes on locally smooth surfaces reconstructed from point clouds", Image and Vision Computing, Vol.25, Issue.1, pp.103-113, 2007.

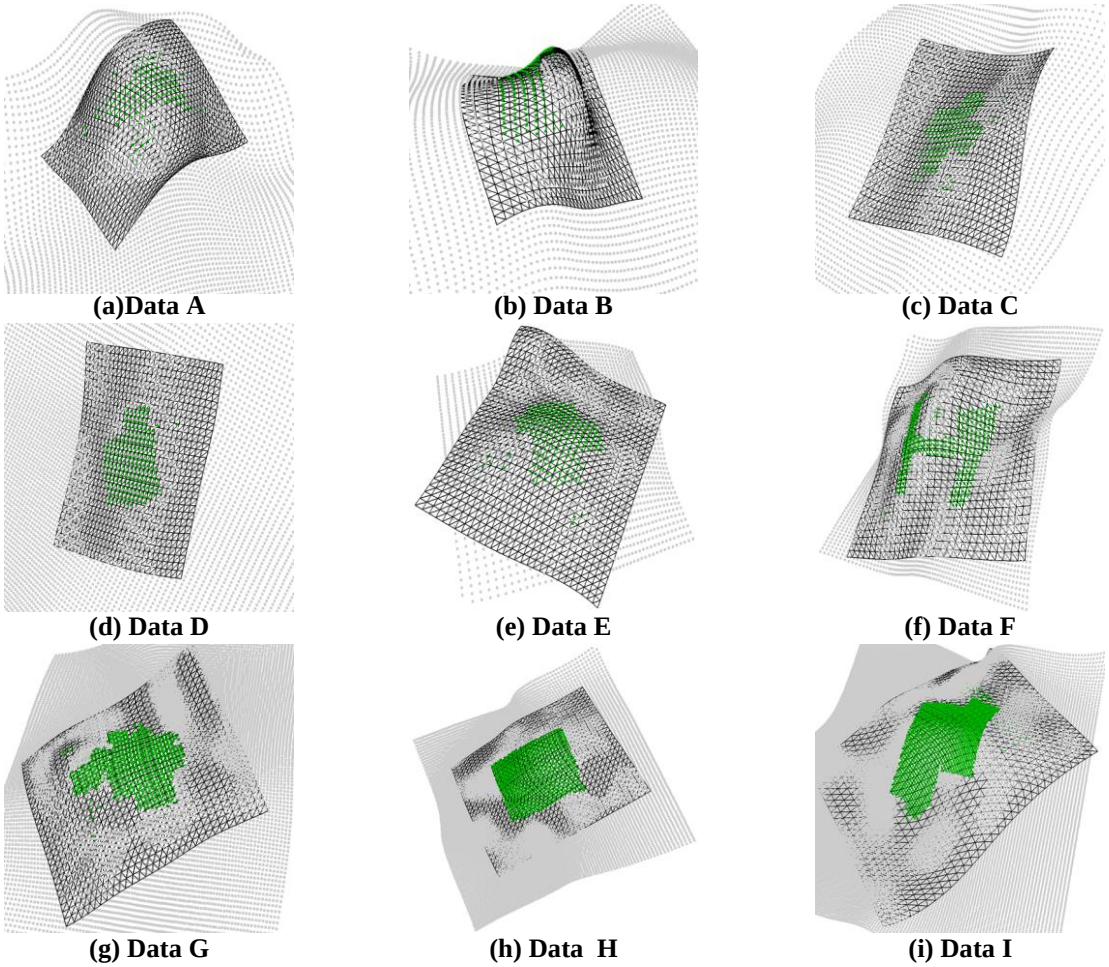
[8] E. Altantsetseg, Y. Muraki, F. Chiba and K. Konno, "3D Surface Reconstruction of Stone Tools by Using Four-Directional Measurement Machine", The International Journal of Virtual Reality (IJVR), Vol.10, No.1, pp.37-43, 2011.

[9] E. Altantsetseg, K. Matsuyama and K. Konno, "Minimum Surface Area Based Complex Hole Filling Algorithm of 3D Mesh", The Journal of the Society for Art and Science Vol.14, No.2, pp.26-35, 2015.

[10] Emelyanov,A., Skala,V., "Surface Reconstruction from Problem Point Clouds", Graphicon 2002, Int.Conf.on Computer Graphics, pp.31-37, 2002.

[11] Emelyanov,A., Skala,V., "A Method for Repaing Triangulations", Graphicon 2004 Int.Conf., pp.180-184, 2004.

[12] Gerhard H. Bendels, Ruwen Schnabe and Reinhard Klein, "Detecting Holes in Point Set Surfaces", The Journal of WSCG, Vol.14(1-3), pp.89-96, 2006.

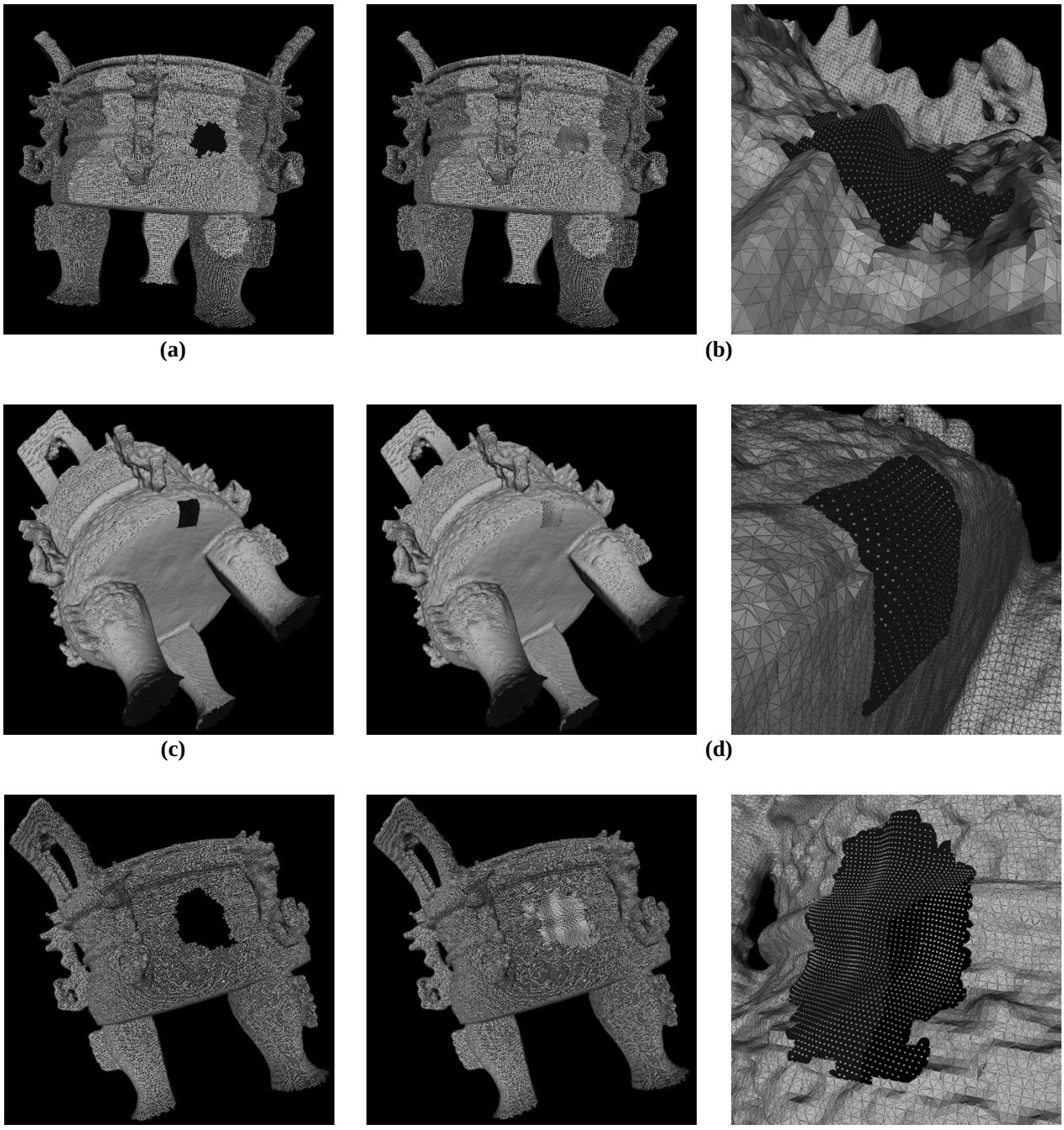


(a)Data A **(b) Data B** **(c) Data C**

(d) Data D **(e) Data E** **(f) Data F**

(g) Data G **(h) Data H** **(i) Data I**

Figure 5. Result of applying to the experimental data



(a) **(b)**

(c) **(d)**

(e) **(f)**

Figure 6. Result of applying to the actual scanned data

Efficient Modeling Methods of Large-scale Model for Monte Carlo Transport Simulation

Guiming Qin
Institute of Applied Physics and Computed Mathematics,
Beijing 100094, China
qin_guiming@hotmail.com

Yan Ma
Institute of Applied Physics and Computed Mathematics,
Beijing 100094, China
ma-yan@iapcm.ac.cn

Yuanguang Fu
CAEP Software Center for High Performance Numerical Simulation, Beijing, China
Fu-yuanguang@iapcm.ac.cn

ABSTRACT

Monte Carlo methods are widely used in simulation of the full-core reactor. They are usually adopted to deal with the geometry model based on the Constructive Solid Geometry. A visual modeling software for the automatic particle transport program, called JLAMT, is developed by the institute of applied physics and computation mathematics. It provides the computing model for Monte Carlo simulation codes, such as Geant4(a software toolkits developed by CERN) and JMCT(a 3D Monte Carlo neutron and photon transport code). To get a better result, the detailed model is needed. For devices as complex as full-core reactors, tens of thousands solids are needed to represent the model. This paper brings up efficient modeling methods of implicit modeling and layer-based modeling for solving this problem. And the effects to the overlap checking are discussed. Taking the full-core reactor of Daya bay power station as an example, experiments show that, by using the efficient modeling methods, both the amount of solids and the time of the overlap checking are reduced.

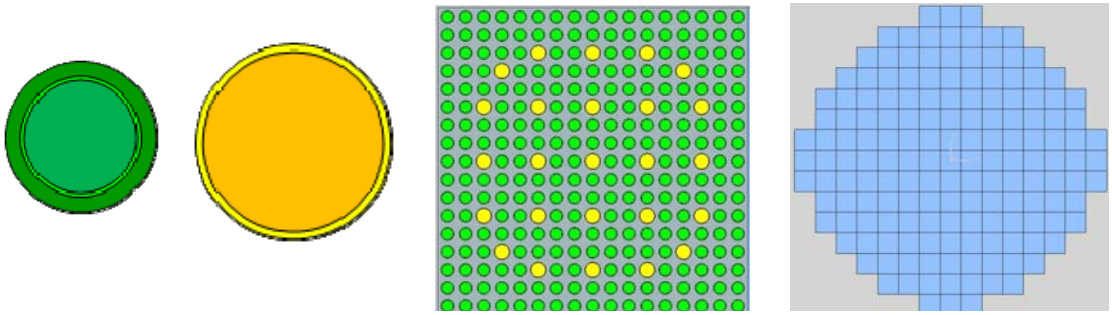
Keywords

implicit modeling, large-scale model, reactor simulation, Monte Carlo.

1. INTRODUCTION

Monte Carlo methods are a broad class of computational algorithms, born in the last century, and widely used to analyze full-core reactor configurations with exact representations of the complex geometry and physical phenomena[Met12] to observe the impact on key reactor parameters like nuclide concentrations, conversion ratio or distribution of generated power concerning, or neutron cross-sections and so on [Kep12, Mar12]. The production codes using Monte Carlo methods describe the geometry usually based on Constructive Solid Geometry (CSG) in Geant4 [Ago03]. A 3D Monte Carlo neutron and photon transport code, called JMCT, is developed for the simulation of the collision of particles with multi-group energy or continuous energy. It also uses CSG to represent geometry. All these codes need a text-based geometry input.

CSG is a solid modeling technique used to allow modelers to create a complex surface or object by using Boolean operators to combine simple objects [Wie05]. The simplest solid objects are called primitives of simple shapes, such as the cuboid, cylinder, sphere and so on.CSG is widely used in computer aided design (CAD), since it has many advantages: 1) modelling using primitives and Boolean operations is much more intuitive than specifying B-rep surfaces directly; 2) the primitives can be associated with additional information [Gui05]. To obtain a simulation with high precision, the geometry models need to be more realistic and detailed. As the computer facilities and capabilities are developed tremendously in decades, geometry modeling capabilities grow from thousands in the 80s to millions now [Bro96]. A text-based geometry description is a tedious, time-consuming and error-prone process, especially for complex geometries of reactors [Wil08]. We have developed visual modeling tool JLAMT for the field application of large-scale model. It relies on the UG CAD kernel, provides functions to build a computing model with geometry representation and physical data, and transforms it to a GDML format file that Geant4 and JMCT can accept.



a)Different cells b) One assembly c) The core of the reactor

Figure 1. Components repeated in a reactor core

A device as complex as a full-core reactor, requires tens of thousands of primitives in CSG. It is a huge pressure for memory and display (may cause lag phase when operating interactively). Even when modifying the whole model, it is hard to observe and pick up. And the more primitives used, the more time-consuming for correctness detection. In this paper, we present methods and strategies to handle and display huge amount of primitives, and build the computing model efficiently. And we discuss JMCT successfully simulated the full-core pin-by-pin problem of Daya Bay Nuclear Power Station Reactor, and calculated the k_{eff} and local tallies with the sophisticated model built by JLAMT

2. Efficient modeling strategy

2.1 Hierarchy Tree and Implicit modeling

The computing model for Monte Carlo simulation includes geometry shape, physical material, and spatial relationship. And a solid may contain a smaller solid in different materials, they have mother-daughter relationship. As a concept by Geant4, a solid with material information are logical volume. A copy of logical volume with position and rotation inside the mother volume creates physical volume [Ago03]. The supported code JMCT uses a GDML format file as model input. GDML (Geometry Description Mark-up Language) is extended from XML with many aspects of a geometry, including material properties and assemblies[Chy06].

The largest volume is called world, and all other volumes are the daughters of the world. For the spatial relation, we can build up a hierarchy tree, with the root of world. In the nuclear devices, many components are repeated with the same geometry shape and material, shown as in Figure 1. One assembly is composed by hundreds of cells, and the core is filled with hundreds of assemblies. As complex as a nuclear reactor core, a device is formed by millions of volumes. We use the implicit modeling strategy to relieve the heavy load for modeling and display. Shown in Figure2, it is a hierarchy tree of physical volumes. Volume A and volume A' are exactly the same, except their positions in space. It means that volume A and A' have the same geometry shape, the same material, and their daughters and descendants are same in shape, material and relative positions with their mothers. When creating the whole model, we create volumes of the sub-tree of volume A, and only draw the solid of volume A' without any details of descendants on the screen. All information of A' is copied from the original volume A. Therefore， the implicit modeling saves the memory and time to model and display.

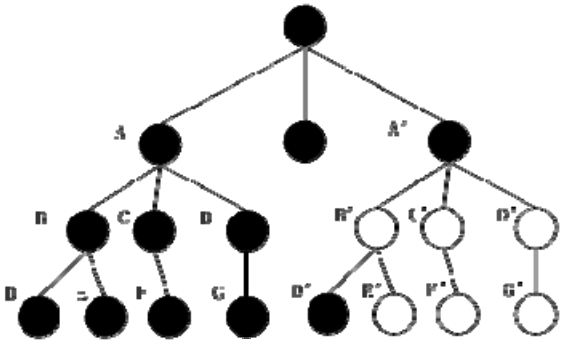


Figure2. The hierarchy tree of physical volumes.

2.2 Overlap checking

Two volumes have mother-daughter relationship if one is fully contained within the other. However, if a volume extends out of the boundaries of its mother volume, defined as overlapping. And if two volumes with no relation intersect, they are also being defined as overlapping, shown in Figure 3.

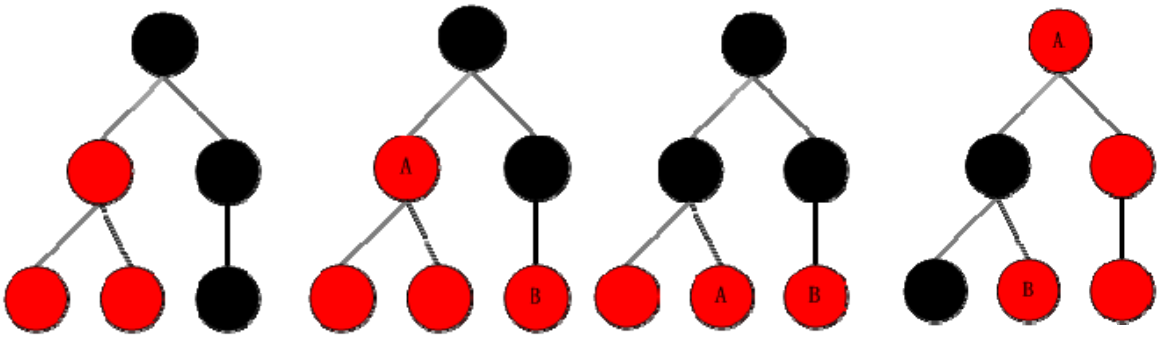


Figure 5. Four situations of volumes occurredin layer-based modeling

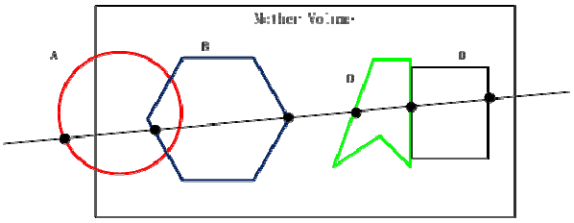


Figure 3.Different cases of volumes overlappingeach other

Detecting overlaps between volumes is bounded by the complexity of the solid model description. Unlike usual detection of CAD, the intersection between two volumes should be checked again if they are mother and daughter. The straightforward detecting algorithm is described by pseudo code below:

```
Function overlapDetect (HierarchyTree node)
    for each child of node
        CAD detect with pair of (child, node)
        if child intersects node
            check intersection with node
            if intersection is beyond the boundary of node
                add ovlapping data to record
            end if
            else add ovlapping data to record
            end if
            overlapDetect(child);
    end for
    for child1 and child2 in node
        CAD detect with pair of (child1, child2)
        if child1 intersects child2
            add overlapping data to record
        end if
    end for
```

The function overlapDetect () is a detecting method of particle transport using one thousand particles to collide with the volume.

Since the volume A' with implicit modeling has the same structure of children as the original volume A, if the original volume does not overlap with its children, A' is correct and vice versa. Therefore, creating only the solid of A' doesn't affect the result of overlap detection.

2.3 Layer-based modeling

A geometry model for complex is composed by tens of thousands volumes. It has huge pressure to display and may cause lag phase when operating interactively. Even when modifying the whole model, it is hard to observe and pick up. The major technique for managing the complex information contained in larger drawing and CAD models is layering [Howard07]. We use layer-based modeling method to narrow down the amount of volumes on the screen.

The main idea is that we create segments of the whole device in separate layers, and display the segments in one particular layer once. We can only create, edit or modify the volumes in the layer which is set to be the working layer. The other volumes not in the working layer are invisible by default. Since all layers share one coordinate system, we combine all segments intuitively as the layers are transparent, and form the whole computing model. The figure4 illustrates the hierarchy tree of the combined model.

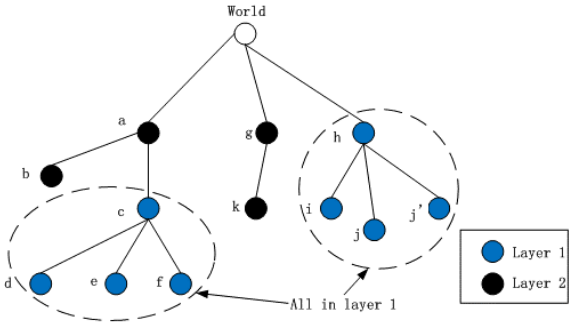


Figure 4. Relationship between hierarchy tree and volumes in layers

The strategy to create volumes in a layer depends on modelers. They can create assemble units separately or volumes with the same material in one layer. We also provide a function that volumes in a layer can be exported as a sub-model for partial simulation. The hierarchy tree of the whole model (complete tree) is separated and a hierarchy tree is built in one layer. There are several situations for the hierarchy tree shown in figure 5, volumes in red are in the same layer. The best one is the case a) where the hierarchy tree of one layer is a sub-tree of the complete tree; cases b) and c) are similar, where the mothers of volumes A and B are not the same, thus we give a virtual mother "world" to them; in case d), volumes A and B in different layers seem independent, but A and B overlap in space, which will cause some problems. Therefore, when doing the layer-based modeling, we check automatically with the rules as follows (taking the schematic in Figure 4 as example):

- The mother-daughter relationship is allowed in one layer (e.g., c and d);
- Volumes in different branches can be in one layer (e.g., c and h);
- Two volumes cannot be in one layer if there are other volumes, which do not belong to the same layer, between (e.g., a and f);
- The volume built with implicit modeling must be in the layer with the original volume (j and j').

2.4 Overlap checking in layers

Volumes in one layer can be exported and be consider as an individual model. Overlap checking for this model is need. Before checking, we preprocess the hierarchy tree. We add a virtual mother Volume "world" to them, and detect whether the situation d) in figure5 is occurred. Then we get a new hierarchy tree of volumes for the exporting layer. With this hierarchy tree, overlapping caused by layer-based modeling is eliminated.

Layer-based modeling can also accelerate the overlap checking. If volumes on one layer are checked, they are marked. When we check the whole model, relations between the marked volumes will not be calculated. If a sub-tree is in the checked layer, the sub-tree can be removed from the whole hierarchy tree, except the root of the sub-tree.

We can also check volumes layer by layer first, then form a new hierarchy tree with roots of sub-trees in the layers. The speed for overlap checking relies on the separation of whole model deeply.

3. Experiment

The full-core reactor of Daya Bay power station is very complex, the core includes 157 fuel assemblies, each component has 17*17 fuel rods, each fuel rod is composed by 2 or 3 units and needs to be divided into 16 sections. There are also many parts outside the core, including detectors, irradiation surveillance capsules, reflecting board, concrete protecting walls and so on. Total number of volumes is 1.46 million of the whole reactor. The model of the full-core reactor is created by both implicit modeling and layer-based modeling methods in figure 6. a) is the assemble with two different cells; b) is the top view of the reactor core with implicit modeling; c) is the reactor core divided into 16 sections; d) is the core with further implicit modeling. The number of total volumes reduced to 6573 with implicit modeling method. It is less than 1% of the original number. We try to reduce it further by adding virtual mother in the core. And we export some parts of the reactor as partial models. The number of volumes and the time for overlap checking are in table 1.Since the model is for nuclear particle simulation, we check the overlap using the detecting method of particle transport without parallel optimization.

	Number of volumes	Overlap checking time
The whole reactor without any methods	1.46 million	3.57 hours
The whole reactor with implicit modeling	6573	48.5 min.
The whole reactor with further implicit modeling	4211	10 min.
The detector	460	5.5 min.
The core with further implicit modeling	1040	2 min.

Table 1.The number of volumes and the time for overlap checking for modeling

We export a GDML file of the whole reactor with implicit modeling. JMCT successfully simulated the full-core pin-by-pin problem of Daya Bay Nuclear

Power Station Reactor with the GDML file, and the visualized result is shown in figure 7.

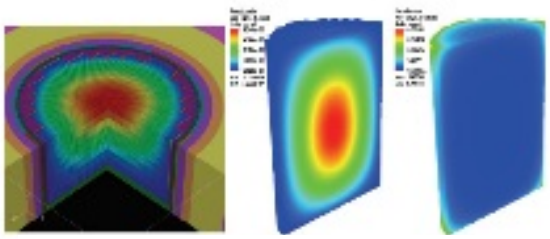


Figure 7. visualized result of the simulation

4. Conclusion

To observe the Monte Carlo simulation of complex devices precisely, such as a full-core reactor, the computing model needs to be realistic and detailed, which is represented by millions of volumes. In this paper, we present the efficient modeling methods, implicit modeling and layer-based modeling. The methods reduce the amount of volumes needed, and he burden on memory and the model displaying. We also discuss the effects for overlap checking that our methods bring to. Implicit modeling does not impact the result of overlap checking, but shortens the computing time. With the layer-based modeling, components of the device can be exported as an independent model.

5. REFERENCES

[Ago03] Agostinelli,S., et al., Geant4 - A Simulation Toolkit, Nuclear Instruments and Methods A 506 250-303, (2003).

[Bro96] Brown, F., Recent Advances and Future Prospects for Monte Carlo, Progress in NUCLEAR SCIENCE and TECHNOLOGY, Vol. 2, pp.1-4 (2011) T. F. Wiegand, Interactive Rendering of CSG Models, Computer Graphics Forum, 15(4):249-261, 1996.

[Chy06] R. Chytracek , McCormick, J., Pokorski, W., Santin, G., 2006. Geometry description markup language for physics simulation and analysis applications. IEEE Trans. Nucl. Sci. 53 (5), 2892–2896

[For 83] Forester, R., Godfrey, MCNP—a general Monte Carlo code for neutron and photon transport, in Methods and Applications in Neutronics, Photonics and Statistical Physics (R. Alcouffe, R. Dautray, A. Forster, G. Ledanois, and B. Mercier, eds.), pp. 33–47, New York: Springer-Verlag, 1983.

[Gui05] Guidera, S., Assessing New Applications for Solid Modeling (Csg) In Architectural Education Paper presented at 2005 Annual Conference, Portland, Oregon. https://peer.asee.org/14527

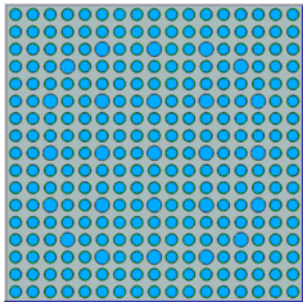
[How07] Howard, R., Bjork, B., Use of standards for CAD layers in building, Automation in Construction, vol. 16, no 3, 290 – 297, 2007

[Kep16] Kepisty, G., Cetnar, J., Stanisz, P., Parametric studies of the PWR fuel assembly modeling with Monte Carlo method, Annals of Nuclear Energy, Vol94,189-207,2016

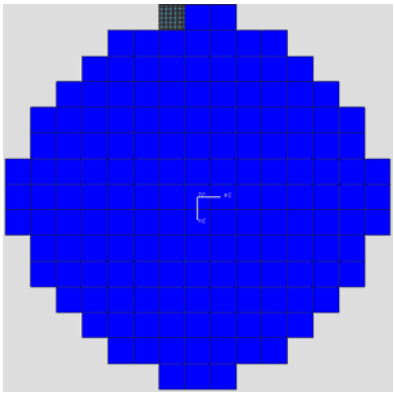
[Mar12] Martin, W., Challenges and prospects for whole-core Monte Carlo analysis, Nuclear Engineering and Technology, V44, no 2, 151-160, 2012

[Met87] Metropolis, M., the Beginning of the Monte Carlo method, Los Alamos Science, Special Issue 1987.

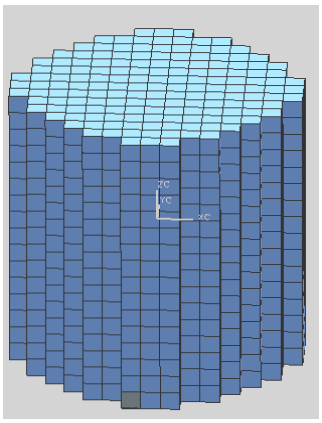
[Wil08] Wilson, P., Feder, R., et al., State-of-the-art 3-D radiation transport methods for fusion energy systems, Fusion Engineering and Design, Vol83, no 7, 824-833,2008



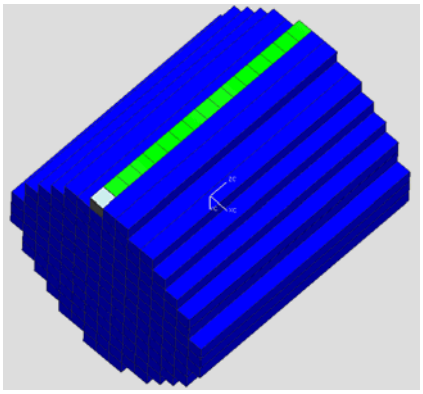
a) The assmbly



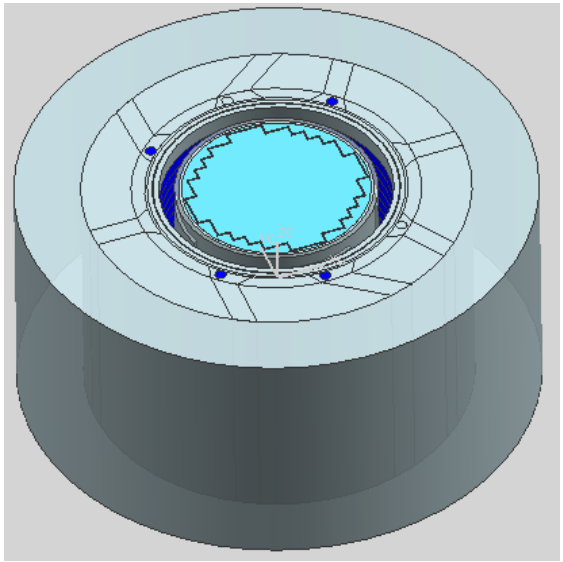
b) The top view of the reactor core



c) the reactor core divided in 16 sections



d) the reactor core with further implicit modeling



e) The complete pin-by-pin model for Daya

Figure 6. Daya Bay full-core pin-by-pin model

Volumetric ultrasound imaging: modeling of waveform inversion

Igor Belykh
St. Petersburg Polytechnic University
Polytechnicheskaya, 29
Russia, 195251, St.Petersburg
Igor.Belyh@cit.icc.spbstu.ru

Dmitry Markov
St. Petersburg Polytechnic University
Polytechnicheskaya, 29
Russia, 195251, St.Petersburg
dmitriym93@gmail.com

ABSTRACT

Ultrasound imaging is widely used in medicine and nondestructive testing. Medical ultrasound volumetric imaging can be considered as a competitive method to X-ray CT and MRI. Recent results in breast ultrasound tomography provide spatial distribution of sound speed and attenuation coefficient based on inverse problem solution in frequency domain. Breast tissue can be characterized as relatively homogeneous medium with possible number of small diagnostically important target inhomogeneities (lesions). This paper is focused on numerical model of complex structure with objects of different sizes and properties that can be met in medicine or in various industrial or scientific cases. High resolution ultrasound imaging is sensitive to a number of effects such as reverberation, diffraction and attenuation that can degrade image quality and cause the artifacts. Those effects are investigated and eliminated by iterative waveform inversion performed in time domain for volumetric visualization of objects' structure and acoustic properties.

Keywords

Visualization, ultrasound imaging, reverberation, diffraction, attenuation, migration, tomography

1. INTRODUCTION

Ultrasound (US) imaging is based on generation of high frequency sound waves in elastic medium and registration of signals propagated through medium interior. Piezoelectric transducers acquisition geometry is designed for solution of specific problem by means of registered wave field processing to obtain a 2-D or 3-D images. Spatial resolution of final image is defined by sounding pulse dominant wavelength and usually varies in a range of fraction mm to few mm. The irregularity of investigation medium in space and physical properties causes elastic waves reflection, refraction, reverberation and attenuation on objects of size comparable or greater than wavelength of sounding pulse [Kak99]. Conventional beam-forming US imaging provides the shapes of investigating objects based on reflected signals forming a 2-D cross sections.

Volumetric US imaging is partially based on main achievements in previous geophysical imaging of the earth interior. One of the known seismic methods [Vir09] is wave form inversion (WFI). It is a qualitative reconstruction method measuring the variations of acoustic impedances originally developed in the 1980s [Tar84] for geophysical imaging of subsurface and sub-bottom [Bel90] objects. Classic WFI approach uses acquired reflection data and forward problem modeling in order to iteratively reconstruct acoustic properties of elastic medium. It can be implemented in frequency domain or time domain. The latter one was not used in practice due to high computational time consuming.

2. RECENT SOLUTIONS

Significant progress in ultrasonic volumetric imaging applied research was achieved during the last decade when various methods of US tomography (UST) methods have been developed and prototyped. Ray based computerized ultrasound tomography (CUT) was advanced by close to practical use diffraction tomography (DT) [Pin05],

[Has13] and waveform tomography [San15]. Transmission US tomography results were compared with X-ray CT and MRI of breast in [Opi13]. Some authors in earlier times [Car81] and later [Dur07], [Hua16] use both reflection and transmission waves for tomographic imaging. Most of the approaches above are focused on medical diagnosis of breast pathologies relying on combination of relatively homogeneous surrounding elastic medium with small inhomogeneous pathologic objects. They use toroidal array of transducers surrounding the target object and scanning it in vertical dimension to obtain a set of slices. The joint reflection-transmission approach [Hua16] uses two parallel linear sensor arrays placed horizontally on both sides of the breast with incremental vertical scan. The most valuable results of these recent solutions are the ability to reconstruct breast structure from reflected component and sound velocity and attenuation distribution from transmitted component using different methods [Lit02]. The disadvantage of transmission tomography methods is that testing medium must have the size less than the size of scanning equipment and must be positioned between source and receiver transducers in order to register propagated waves. Such a limitation reduces the use of described methods in technical applications where sizes of objects may vary tremendously. We propose to use the volumetric US reflection inversion for visualization of complex structures containing objects of different sizes and physical properties for medical or nondestructive testing purposes. Here we meet problems with diffraction, reverberation and attenuation effects.

Diffraction is one of the main problems for US imaging. Without diffraction UST would be similar to X-ray tomography. Diffraction in breast US imaging was investigated in [Sim08] concluding better spatial resolution of diffraction tomography compared to transmission methods. Diffractions can be eliminated with migration method as shown in [Sch11] and [Gar13] developed in time and frequency domains respectively for medical application.

One of the efforts close to practical use in medicine was recently presented by [San15]. Authors use WFI principles to reconstruct sound speed in breast tissue based on Helmholtz equation solution in frequency domain. The restored images with sound speed mapping are of good quality but attenuation problem still needs to be solved.

A discussion on the pros and cons of WFI time-domain versus frequency-domain modeling [Vir09] is still open. One obvious advantage of frequency domain is computational efficiency but its disadvantage is noticeable spatial oscillations in reconstructed images. Time domain inversion despite its low efficiency provides good spatial resolution. Another its advantage is easy separation of reflection and transmission components while solving the forward problem.

The goal of this paper is to apply reflection US imaging for complex structure interior volumetric visualization.

3. PROPOSED SOLUTION

The elastic medium structure and properties are modeled and then reconstructed using the iterative reflection wave field inversion approach for a stratified elastic medium in time domain. The solution of forward and inverse problems is decomposed into several steps using both ray and wave theoretical foundations at different stages in order to investigate and eliminate (or compensate) the most critical effects that degrade image quality separately.

3.1 Forward problem

This paper presents the inversion technique by numerical modeling of the most energetic normal incident part of the field. This is provided by two assumptions: (1) emitter and receiver have zero offset in the same transducer, (2) all objects in the model are formed by planes parallel to coordinate planes as shown in Fig.1.

The proposed numerical model simulates a complex structure with objects of different shape, size and properties that can be analogous to various parts of technological systems in nondestructive testing or to abdominal body parts in medicine. The model structure (Fig.1) represents a homogeneous liquid medium of size 300x200x200 mm containing three objects inside.

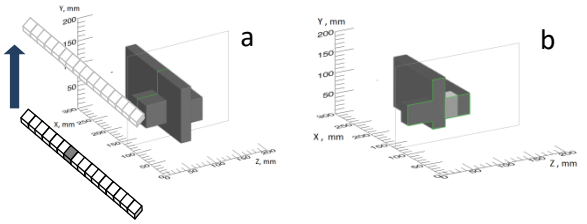


Figure 1. Numeric model and acquisition scheme: (a) model geometry and sliding transducer array forming normal incident wave field, (b) – numerical model cut by *(x=130, y, z)* plane shown on Fig.1a.

Large object of soft tissue or material has maximal sizes about 190x120x90 mm and incorporates a medium size object in the form of a channel filled out with acoustically contrast substance (solid tissue or material). Small object of size 4x4 mm simulates a lesion or debris and is located at a point with coordinates *(x=50, y=270, z=80)*.The shape of large object is complex but is formed by parallel planes in all three dimensions. This simplification allows to investigate vertically inhomogeneous model at each transducer position using one dimensional wave equation. The imaginary array of transducers is located on *(x,y)* plane and moves along the surface of liquid medium as schematically shown in Fig.1a. Thus we consider only compressional waves without shear and surface waves. In order to investigate only the normal incident part of the reflected pressure field *p(x,y,z,t)* at the medium surface *(x,y,z=0)* the transducer incrementally moves along the array in *x* direction (Fig.1a). The array scans the model moving along y direction in *(x,y)* plane with a step of 1 mm. Each element of transducer array is activated sequentially (as highlighted in gray) to send a pulse and to receive the reflected signals. Thus a virtual matrix of emitters insonifies the whole structure with normal incident plane waves in *z* direction. This acquisition scheme is similar to one used in [Hua16].

The distribution of velocities of sound waves, densities and attenuation coefficients of modeled objects are shown in Table 1.

Reverberation effect has the most influence in registered field at reflection/refraction angles close to normal.

Type of material	Velocity [m/s]	Density [g/m^3]	Attenuation coefficient [1/cm]
Liquid	1480	1.0	0.001
Soft tissue/ material	1510	1.05	1.0
Solid tissue	4080	1.9	2.0
Lesion/debris	2500	1.4	1.5
Air	331	1.29	1.26

Table 1. Acoustic properties of different objects used to form the model in Fig.1.

The problems of reverberation in conventional B-mode reflection ultrasound modality and method for its suppression were described in [Shi16]. Similar problem can be clearly demonstrated on a suggested model with parallel interfaces and zero-offset acquisition technique for reflected waves. Pressure wave propagation through the presented model along each emitter-receiver ray can be obtained using the finite-difference method for one-dimensional wave equation solution in case of impulse source. Reflectivity function $r(z)$ is obtained using acoustic impedance $\gamma(z) = \rho(z)V(z)$ distribution computed from Table 1. Reflection coefficient for air-liquid interface is $r(0) = 0.9994$. Next we substitute the spatial variable z with one-way travel time τ along the ray defined as

$$\tau(z) = \int_0^z \frac{dz}{V(z)} \qquad (1)$$

Then we model pressure field $p(\tau, t)$ by acoustic wave equation solution with appropriate boundary conditions using bent-ray approximation:

$$p_{\tau\tau} + [\ln \gamma(\tau)]' p_\tau = p_{tt} \qquad (2)$$

The solution of equation (2) provides full wave field including all reverberations.

Diffractions blur inhomogeneities reducing spatial resolution on reflected US images [Has13] thus, for example, forming speckles.

Since ray approach does not presume diffraction effect [Kak99] we compute diffractions from all of inhomogeneous points in 3-D space based on fundamental principle of Huygens' secondary source. The response in (x, y, t) space from such a source forms a diffraction hyperboloid according to:

$$t(x,y) = \sqrt{t_0^2 + \frac{(x-x_0)^2+(y-y_0)^2}{v^2(x_0,y_0,t_0)}} \qquad (3)$$

where x_0, y_0, t_0 – are coordinates of diffraction point, and v is sound velocity. In 3-D we receive hyperboloids at each point of diffraction. The edges and corners of the model are blurred with diffractions as can be seen on superposed reverberation and diffraction field shown in Fig. 2.

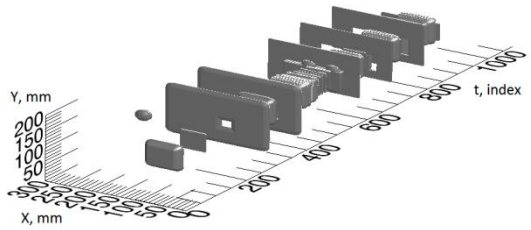


Figure 2. Reflected wave field including reverberations and diffractions.

Computational equivalent of δ-function used for modeling in previous steps is an ideal approach but to model a physical process we need to take in account a transducer transfer function (TF). A typical piezo element TF can be modeled as described in [Myl07]. In order to use reflected waves amplitudes in inversion we need to register signals in the far field. We can estimate the near field distance as $\frac{D^2}{4\lambda}$, where D is transducer diameter and λ is generated wavelength. For liquid material used in the model of this paper and for 1 MHz pulse and D=10 mm this distance is about 17 mm which is suitable for acquisition geometry we describe. Sounding pulse similar to one in [Myl07] has central frequency about 1 MHz. The superposed field with multiple reflections and diffractions is convolved with pulse described above.

Attenuation was modeled as exponential coefficient applied to amplitudes of waves while they travel up and down the model interior including multiples according to equation (2):

$$a(t) = s(t)\, e^{-bt} \qquad (4)$$

where b – is attenuation coefficient from Table 1, in which b values for medical application correspond to water, bone, soft tissue and lesion. Dispersion is not considered as it is small in soft tissue [Kak99].

Despite the influence of attenuation reverberations cause artifacts in the form of dominating false reflectors.

Only the first and some reverberated reflectors and diffractors can be observed on final images due to attenuated energy. Small object usually seen as a speckle in regular US B-forming scan forms a hyperboloid in 3-D (Fig.2).

To reconstruct the model we need to solve inverse problem.

3.2 Inverse problem

We make inversion of final reflected wave field in few steps in order to restore the model structure.

At first step, we need to deconvolve the registered field $a(x,y,t)$ by transducer transfer function. Since each transducer transfer function $H(f)$ can be measured precisely the convolutional integral equation can be solved in frequency domain to receive a deconvolved impulse response $g(t)$ and can be embedded into equipment to be applied in-vivo:

$$\mathrm{g}(t) = \boldsymbol{F^{-1}}[G(f)] = \boldsymbol{F^{-1}}\left[\frac{A(f)H^*(f)}{|H(f)|^2+\alpha^2}\right] \qquad (5)$$

where $A(f), G(f)$- are Fourier transforms of registered and deconvolved signals, $\boldsymbol{F^{-1}}$ denotes inverse Fourier transform , $\alpha = \alpha(f)$ regularization parameter practically depending on noise level and its frequency composition [Bel90]. At US frequencies mainly electronic noise affects the signals so α can be taken as a standard deviation of equipment self-noise. The main role of parameter α in present model is to compensate spectrum distortions by attenuation. Deconvoled pulse contains some noise produced by non- ideal inverse filtering.

The next step is diffraction artifacts elimination in 3-D space. According to [Kak99] the first Born approximation is valid only when the scattered field is smaller than the incident field. Since absolute values of reflection coefficients are smaller than 1 total reflected and reverberated energy at receiver is less than it was generated by source. Diffraction

tomography assumes that scattering arises from incident field only.

Diffraction effect is modeled based on the principle that each inhomogeneous point of medium is a Huygens' secondary source with angle-dependent point aperture for all types of waves including reverberations. So we can use migration procedure of pressure wave front to suppress all diffractions in $g(x, y, t)$ space. One of the robust migration methods is based on Kirchhoff integral [Sch11]. For practical use the term proportional to $\frac{1}{R^2}$ is omitted [Yil87]. In order to get output field $g_{out}(t)$ for any output spatial point (x, y, z) from input field $g_{in}(x_{in}, y_{in}, z, t)$ we can use it in the following form:

$$g_{out}(x, y, z, t) =$$

$$= \iint dx\, dy\, \frac{\cos(\theta)}{vR} \frac{\partial}{\partial t} g_{in}(x_{in}, y_{in}, z, t - R/v) \quad (6)$$

where $R = \sqrt{(x - x_{in})^2 + (y - y_{in})^2 + z}$, $\cos(\theta) = 0$ for presented model with parallel interfaces and normal incident waves. Migration procedure is applied to each point of registered wave field to obtain the migrated field $\hat{p}(x, y, t) = g_{out}(x, y, z = 0, t)$ at the receiving surface.

The reverberations from acoustically contrast objects dominate the rest of informative reflections and need to be suppressed. The inverse finite-difference scheme for equation (2) deploys a backpropagation algorithm providing all reverberation suppression and reflectivity function $\hat{r}(\tau)$ restoration from deconvolved and migrated impulse response $\hat{p}(t)$.

The most critical step in inversion process is attenuation correction. The exponential correction reverse to (4) is applied with adjusted power to compensate the attenuated reflectivity. The exponent power parameter can be estimated as a product of double depth of ultrasound penetration, presumed averaged attenuation coefficient and tuning coefficient which was obtained based on the limit for the maximal reflection coefficient value $|\hat{r}(\tau)| \leq 1$. Tuning coefficient was found equal to 0.75 as an optimal value for the case of presented model (Fig. 3). Such function can be designed in real systems with sophisticated algorithms for adjustment by operator while imaging process.

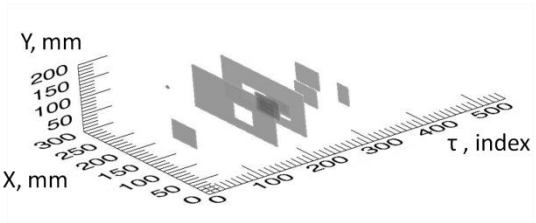


Figure 3. Reflection wave field after inversion and attenuation correction.

Acoustic impedance distribution can be obtained from restored reflectivity using [Old83] :

$$\hat{\gamma}(\tau) = \exp\left(\int_0^{\tau} 2\hat{r}(\vartheta) d\vartheta\right) \quad (7)$$

The restored impedance is a function of one-way travel time τ. The restored and initial arrival times have good correlation but absolute values for acoustically contrast internal object differs a lot mainly due to strong influence of attenuation especially in solid tissue or hard material of internal medium size object.

In suggested inversion approach the impedance is restored as a function of one-way travel time and to get it in space we need to split the product of density and velocity. Velocity function provides the main contribution to this product, so we indirectly receive a velocity distribution. Since the exact velocity distribution is known for the presented model the impedance is scaled accordingly to be presented as a spatial 3-D function to compare it with original model (Fig.1).

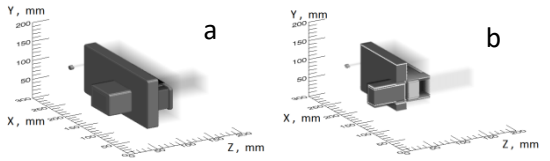


Figure 4. Restored model: acoustic impedance volumetric distribution as a spatial function (a), the same as (a) cut by *(x=130, y, z)* plane.

Computational noise at all steps of inverse problem resulted in some distortions in restored impedance model as can be seen in Fig.4. The edges of the large object were smoothed while migration step. Despite the distorted impedance values the objects of all sizes can be revealed in shapes close to initial ones.

4. CONCLUSIONS

An effort was made to volumetrically visualize the objects of different sizes and shapes with quantitative estimate of their acoustic impedances. This result was based on time domain inversion of US reflected field registered on the surface of vertically inhomogeneous elastic medium insonified by normal incident plane waves. Deconvolution and migration were applied to improve the spatial resolution of the final image in the form of acoustic impedance function of two spatial coordinates and one-way travel time. Sound velocity distribution can be estimated from impedances that may lead to present final structure volumetrically. Attenuation is a difficult effect to compensate by suggested inversion approach.

5. REFERENCES

[Bel90] Belykh, I. N., Inversion of seismoacoustic sounding data. In Russian geology and geophysics, Vol. 3, pp 105-111., (in Russian with abstract in English), 1990.

[Car81] Carson P. L., Meyer C. R., Scherzinger A. L., and Oughton T. V., "Breast imaging in coronal planes with simultaneous pulse echo and transmission ultrasound," In Science, Vol. 214, p. 1141-1143, 1981.

[Dur07] Duric N., Littrup L., Poulo P., Babkin A., Pevzner R., Holsapple E., Rama O., Glide C.. Detection of breast cancer with ultrasound tomography: First results with the Computed Ultrasound Risk Evaluation (CURE) prototype. In Med. Phys., Vol. 34, p. 773-785, 2007.

[Gar13] Garcia, D., Le Tarnec, L., Muth, S., Montagnon, E., Porée, J., Cloutier, G. Stolt's f-k. Migration for Plane Wave Ultrasound Imaging. In IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 60, No. 9, p.1853-1865, Sep. 2013.

[Has13] Hasegawa, H., Kanai, H. Comparison of Spatial Resolution in Parallel Beamforming and Diffraction Tomography. In Proceedings of Symposium on Ultrasonic Electronics, Vol. 34, p. 315-316. Nov. 2013.

[Hua16] Huang L., Shin J., Chen T., Lin Y., Gao K., Intrator M., Hanson K. Breast ultrasound tomography with two parallel transducer arrays. In Medical Imaging 2016: Physics of Medical Imaging, Proc. of SPIE, 97830, Vol. 9783, Mar. 2016.

[Kak99] Kak, A.C., Slaney, M. Principles of computerized tomographic imaging. New York: IEEE Press, 1999.

[Lit02] Littrup, P., Duric, N., Leach Jr., R. R., Azevedo, S. G., Candy, J. V., Moore, T., Chambers, D. H., Mast, J. E., and Holsapple, E. T., "Characterizing tissue with acoustic parameters derived from ultrasound data". In SPIE: Ultrasonic Imaging and Signal Proceesing, Insana, M. and Walker, W. F., eds., Proc. SPIE 4687, pp 354–361, 2002.

[Mue16] Mueller, P., Schuermann, M., Guck, J. The Theory of Diffraction Tomography. In press. 2016. Available from: https://arxiv.org/pdf/1507.00466.pdf.

[Myl07] Mylvaganam, J. Characterization of medical piezoelectric ultrasound transducers using pulse echo methods. M.S. thesis, Norwegian University of science and technology, 2007.

[Old83] Oldenburg, D.W., Scheuer, N., Levy, S. Recovery of the acoustic impedance from reflection seismograms. In Geophysics, Vol. 48, p. 1318-1337, 1983.

[Opi13] Opieliński, K., Pruchnicki, P., Gudra, T., Podgórski, P., Kraśnicki, T., Kurcz, J., Sąsiadek, M. Ultrasound Transmission Tomography Imaging of Structure of Breast Elastography Phantom Compared to US, CT and MRI. In Archives of Acoustics, Vol. 38, No. 3, p. 321–334, 2013.

[Pin05] Pintavirooj C., Jaruwongrungsee K., Withayachumnankul W., Hamamoto K., Daochai S. Ultrasonic Diffraction Tomography: The Experimental Result. In proc. WSCG, Poster proceedings, 2005.

[San15] Sandhu G.Y., Li C., Roy O., Schmidt S., Duric N. Frequency domain ultrasound waveform tomography: breast imaging using a ring transducer. In Phys. Med. Biol., Vol. 60, p.5381–5398, 2015.

[Sch11] Schmidt, S., Duric, N., Li, C., Roy, O., Huang Zh.-F. Modification of Kirchhoff migration with variable sound speed and attenuation for acoustic imaging of media and application to tomographic imaging of the breast. In Medical Physics, Vol. 38, No. 2, p.998-1007. Feb. 2011.

[Shi16] Shin, J., Chen, Y., Malhi, H., Yen, J.T. Ultrasonic Reverberation Clutter Suppression Using Multiphase Apodization With Cross Correlation. In IEEE Transactions On Ultrasonics, Ferroelectrics, And Frequency Control, Vol. 63, No. 11, p. 1947-1956., Nov. 2016.

[Sim08] Simonetti, F., Huang L., Duric N. and Littrup P. Diffraction and coherence in breast ultrasound tomography: A study with a toroidal array. Technical report 08-4616, Los Alamos National Laboratory, 2008.

[Tar84] Tarantola, A., Inversion of seismic reflection data in the acoustic approximation: In Geophysics, vol.49, p. 1259–1266. 1984.

[Vir09] Virieux J. and Operto S. An overview of full-waveform inversion in exploration geophysics In Geophysics, Vol. 74, No. 6, Nov.-Dec. 2009.

[Yil87] Yilmaz, O. Seismic data processing. In Investigations in geophysics: Vol. 2, 1987.

Convolutional Neural Network Based Chart Image Classification

Jihen Amara
Digital Research Center of Sfax (CRNS), MIRACL Laboratory
University of Sfax, Tunisia
amarajihen@gmail.com

Pawandeep Kaur
Heinz-Nixdorf Chair for Distributed Information Systems, Friedrich-Schiller-Universität Jena, Germany
pawandeep.kaur@uni-jena.de

Michael Owonibi
Heinz-Nixdorf Chair for Distributed Information Systems, Friedrich-Schiller-Universität Jena, Germany
michael.owonibi@uni-jena.de

Bassem Bouaziz
Higher Institute of Computer Science and Multimedia, MIRACL Laboratory
University of Sfax, Tunisia
bassem.bouaziz@isims.usf.tn

ABSTRACT

Charts are frequently embedded objects in digital documents and are used to convey a clear analysis of research results or commercial data trends. These charts are created through different means and may be represented by a variety of patterns such as column charts, line charts and pie charts. Chart recognition is as important as text recognition to automatically comprehend the knowledge within digital document. Chart recognition consists on identifying the chart type and decoding its visual contents into computer understandable values. Previous work in chart image identification has relied on hand crafted features which often fails when dealing with a large amount of data that could contain significant varieties and less common char types. Hence, as a first step towards this goal, in this paper we propose to use a deep learning-based approach that automates the feature extraction step. We present an improved version of the LeNet [LeCu 89] convolutional neural network architecture for chart image classification. We derive 11 classes of visualization (Scatter Plot, Column Chart, etc.) which we use to annotate 3377 chart images. Results show the efficiency of our proposed method with 89.5 % of accuracy rate.

Keywords

Chart Image Classification, Data Visualization, Deep Learning, Dataset Annotation.

1 INTRODUCTION

In recent years, there has been growing interest in data visualization due to its ability to present meaningful insights into the complex and data of ever increasing sizes. Thus, Visualization is important for scientists as it helps them in exploring, analyzing, and publishing their results.

Most of the different types of visualizations and graphs, produced in papers, are designed to be human understandable and are not typically machine readable. In most cases, these visualizations (which are typically in raster image format) are hard to decode by machines. However, there is a growing need in many applications and domains for machine's ability to read, extract and interpret insights presented in chart images automatically. The major challenge when attempting to interpret such charts automatically is the variant structure and visual appearance of these images. Thus, it is difficult for machines to automatically classify the chart type and understand the encoded information efficiently. The field of Document Analysis and Recognition addresses this problem by using Computer Vision techniques to decode the information from charts. This has led to the emergence of a sub field which is known as 'Chart Image Analysis'. Techniques in 'Chart Image Analysis' extract, classify and interpret charts to provide valuable information. This information then helps in chart redesigning, knowledge mining and domain-based visualization recommendation studies.

Motivated by advances in pattern recognition techniques, especially convolutional neural networks (CNNs), which has managed to produce remarkable results in the field of image classification, we propose a new system based on CNNs for chart analysis. The model can learn visual features directly from images and is able to recognize eleven types or classes of charts (Categorical Boxplot, Column Chart, Dendrogram, Heatmap, Line Chart, Map, Node-Link Diagram, Ordination scatterplot, Pie Chart, Scatterplot and Stacked Area Chart). While the identification of a visualization class does not on its own generate the full information that the visualization presents, it can provide basic

and important clues about the representational goal of the statistical graph. For example, whether the graph is representing a comparison among data entities (variables) or is showing the distribution of some entities (variables) over the temporal scale, etc. This information can then be combined with some auxiliary information (e.g. knowledge extracted from carrying out Natural Language Processing on the caption of the statistical graph) to derive the concrete message that visualization conveys.
The rest of the paper is organized as follows. In Section 2, we present the motivation behind this work. The literature review is presented in Section 3. Our proposed method is described in Section 4. Experimental evaluation and results are reported in Section 5. Finally, Section 6 concludes the paper and provides an outlook to future work.

2 MOTIVATION

The motivation behind this work is to automatically classify the chart image types from an image dataset and to use this classification in tagging captions from these chart images. This will help in the creation of domain (biodiversity in our case) knowledgebase which will provide the contextual information about the dataset that needs to be visualized. This further will be an assistant to our visualization recommendation engine. Visualization recommendation is a subdomain in data visualization research that uses different strategies to assist user in the selection of suitable visualization for representing their data. With the growing amount of data and increasing availability of different visualization techniques, the selection of suitable visualization become more critical especially for users who are not well versed in visualization creation process. Hence, visualization recommendation strategies can effectively assist the user in making a choice of an appropriate visualization by considering different aspect of data, domain, user and goals. To provide visualization suggestions for the specific domain, it is important to first understand the visualization usage pattern from this domain. To fulfil that, we have gathered a dataset of 96837 images from 26588 biodiversity publications. To automatically classify this big dataset of chart images into different visualization classes or chart types, we used a model based on Convolutional Neural Network (CNN).

3 LITERATURE REVIEW

Chart image classification is an important step in chart recognition and understanding. One of the major challenges in chart image classification is the variability of the structure and visual appearance of each chart type. To address this challenge, techniques from image processing, raster to vector conversion, layout and pattern analysis have been used. These techniques are broadly divided into model based approaches and machine learning based approaches.

Model Based Approaches

A distinctive model is designed for each chart type that constitutes of specific features of the chart. These features include: (1) graphical elements of the charts, e.g., axis, colors etc. [Shao 05], (2) layout of the chart, e.g., rectangular (for xy charts), circular (for pie charts) etc. [Yoko 97], and (3) patterns or local structures appearing frequently in charts [Inok 00]. By extracting the feature information, a model is developed for each chart type that contain specific feature information about the specific chart and governs some rules and constraints to distinguish one chart from other. The main drawback of the model based approaches is that such a system can handle only predefined charts for which the graphical model was available. Even a slight change in the chart style will be considered different and would need a new model. Thus, such technique cannot be used to classify wide styles of chart. Therefore, machine learning techniques that considers graphical elements of charts as whole object instead of different segments were adopted.

Machine Learning Based Approaches

We have done a survey of the different machine learning techniques used for recognition and classification. Table 1 summarize some work.

Paper	Year	Dataset	# Classes	Technique	Accuracy
[Kart 12]	2012	155	8	KNN	78.06%
				MLP	69.68%
				SVM	69.68%
[Savv 11]	2011	2601	10	Multiclass SVM	80%
				Binary SVM	96%
[Pras 07]	2007	653	5	Multiclass SVM	84%
[Huan 07]	2007	200	4	C4.5 Decision Tree	90%
[Shao 05]	2006	36	5	MLP	94.2%
				LogitBoost	100%
[Zhou 01]	2001	350	3	HMM	86%
				Feedforward Neural Network	77.7 %

Table 1: Machine learning based approaches

Even though different methods have achieved good classification results for identifying and recognizing certain types of chart images, they mostly rely on hand-crafted features such as color histograms, texture features, shape features and Scale-invariant feature transformation (SIFT). Also, these methods do not generalize well and they are not effective when dealing with a large amount of data that could contain significant varieties as in the case of our chart image dataset. For example, the bar chart and column chart are considered the same but their orientation is different. Furthermore, the problem of multi-chart images with same or different chart types hinders the accurate detection. Recently, [Lee 15] have proposed an algorithm to automatically segment multi-chart visualizations into a

set of single-chart visualizations. Besides, due to the lack of standardization in chart construction and chart templates, there exist great diversity of chart types and styles, which are somehow very subjective. Moreover, the accurate detection becomes more difficult when handling degraded, distorted or rotated charts with different scales, noisy and hand-drawn inputs.
Hence, in our work we designed a convolutional neural network (CNN) based model to automatically classify chart images and consequently avoid segmentation and hand-crafted features extraction.

4 PROPOSED METHOD

As described above, different approaches have been proposed in the literature to deal with the problem of chart image classification, the variation in chart style, size, color, resolution and content are not yet resolved (i.e. marks within data graphics are discrete and areas are in uniform color). Hence, following the successful application of convolutional neural networks [Deng 14, Arel 10] in many computer vision tasks, we propose a method based on CNN to classify chart images into eleven types. More specifically, we designed a CNN based model inspired from the LeNet architecture [LeCu 89]. It has a total of eight layers comprising one initial input layer, five hidden layers followed by one fully connected layer and ending with the output classifier layer. The hidden layers are convolution and pooling layers which act as feature extractors from the input images while the fully connected layer acts as a classifier. In fact, the main purpose of convolution is to extract features automatically from each input image. The dimensionality of these features is then reduced by the pooling layer. At the end of the model, the fully connected layer with a softmax activation function makes use of the learned high-level features to classify the input images into predefined classes.

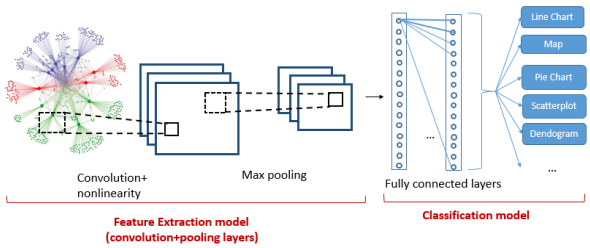


Figure 1: A graphical depiction of the proposed approach

Consequently the proposed model, illustrated in Figure 1, is composed of two main parts: the first part is the self-taught feature extraction model and the second part is the classification model. In the rest of this section, we will detail these two components.

4.1 Feature extraction model

The feature extraction model is the part where the network learns to detect different high-level features from the input images. It consists of a sequence of convolution and pooling layers.

Convolution map: The convolution layer is an elementary unit in a CNN architecture. The goal of convolution is to extract features from the input image. It consists of a set of learnable filters. Each filter is applied to the raw pixel values of the image taking into account the red, green and blue color channels in a sliding window fashion, computing the dot product between the filter pixel and the input pixel. This will result in a 2-dimensional activation map of the filter called feature map. Hence, the network learns filters (i.e., edges, curves) that will activate when they find known features in the input.The CNN learns the values of these filters on its own during the training process.
The Convolution operation is presented in Equation 1. A convolution layer is configured by the number of convolution maps it contains M_i, the size of the filters which are often squared $k_x \cdot k_y$. The feature map M_i is computed as follows:

$$M_i = b_i + \sum_k W_{ik} \star X_k \tag{1}$$

where $\star$ is the convolution operator, X_k is the k^{th} input channel, W_{ik} is the sub kernel of that channel and b_i is a bias term. In other words, the convolution operation being performed for each feature map is the sum of the application of k different 2D squared convolution features plus a bias term. Hence, In comparison with traditional image feature extraction that relies on crafted general feature extractors (SIFT, Gabor filter, etc), the power of CNN is noted in its ability to learn the weights and biases of different feature maps which lead to task specific powerful feature extractors.

Moreover, Rectified nonlinear activation function (ReLU) is performed after every convolution. ReLU is a very popular activation function which is defined as $f(x) = max(0,x)$ where x is the input to a neuron. areIts role is to introduce nonlinearity to the CNN.

Max-pooling map: In the architecture of convolutional neural networks, convolution layers are followed by subsampling layers. A layer of sub-sampling reduces the size of the convolution maps, and introduces invariance to (low) rotations and translations that can appear in the input. A layer of max-pooling is a variant of such layer that has shown different benefits in its use. The output of max-pooling layer is given by the maximum activation value in the input layer over sub windows within each feature map. The max-pooling operation reduce the size of the feature map.

4.2 Classification model

Within the classification step we use fully connected layers where each neuron provides a full connection to all learned feature maps issued from the previous layer in the convolution neural network. These connected layers are based on the softmax activation function in order to compute the class scores. The input of the softmax classifier is a vector of features resulting from the learning process and the output is a probability that an image belongs to a given class. The softmax function ς takes as input a C-dimensional vector z and outputs a C-dimensional vector y of real values between 0 and 1. This function is calculated as below:

$$y_c = \varsigma(\mathbf{z})_c = \frac{e^{z_c}}{\sum_{d=1}^{C} e^{z_d}} \quad for\ c = 1, \cdots, C \qquad (2)$$

In the next section, we will present the conducted experiments and results.

5 EXPERIMENTS

In this section, we evaluate the proposed method on our collected dataset for image chart classification. Experiments were conducted on an Intel Xeon CPU E5 PC with 32 GB of RAM. In our implementation we used deeplearning4j the open source deep learning library in Java. We start by describing the dataset then we present the performance of our approach before comparing obtained results to relevant proposed method from the literature.

5.1 Datasets

We have collected our data set from images presented in biodiversity publications. We have downloaded all available volumes and issues using a Python script from seven different biodiversity journals. This resulted in 26588 publications downloaded, out of which we were able to extract 96837 images. This corpus of 96837 images have mixed content. The majority of the images are related to data visualization and the rest are general images such as camera clicked pictures, conceptual diagrams and flowcharts.
To annotate our dataset we have used the Viper GT tool which is a Java based Ground Truth Software. This tool allows the user to mark up a media file (image file in our case) with the information about its content. This information is then stored in an xml format. So each image has a respective xml file defining the class label of the marked content in the image file.

5.1.1 Annotation Process

In the beginning, we were not aware of different chart types available in the collected dataset. Hence, we created the classes as they appeared in the images and have labeled them. Following this procedure, we ended up with 51 different classes from 1000 images. There was a huge variation in the samples' count per class. To solve this problem of unequal class distribution and to gather more training data, we did another round of annotation. This time, we preselected 11 different classes to annotate this dataset. Then, we had randomly selected 3377 images for our training dataset.

These 11 classes or superclasses were derived according to two important criteria that we have observed from our previous annotation phase: count of images per class and visualization shape similarity. Classes or chart types which are associated with a significant amount of images like Ordination Scatterplot, Map and Line Chart were retained. However, those who had less than 5 images were excluded from further processing unless they have a visual shape similarity to other retained classes. Hence, chart types that use the same coordinate space (e.g. xy plot) and same visual marks (e.g. bars) can be considered visually similar and then merged into one super class. For example, all the chart types which are visually similar to Column Chart e.g. Bar Chart, Stacked Bar Chart, Muli-set Bar Chart and Dual Axis Bar Chart are all merged into one super class as 'Column Chart'. Besides, to increase the number of images per class, we have included additional images for classes whose count was less than 200. These images were gathered from the dataset provided by [Savv 11] and from the Internet. In Table 2, we have presented our final training set count for each class.

5.2 Results

To validate the performance of the proposed approach, we conducted a set of experiments using our collected and annotated dataset containing 3377 images. Our goal is to evaluate the predictive performance of our model on unseen data of chart images. Hence, in our experiments, 80% of the annotated dataset is used for training and the rest 20 % is used for testing with the use of hyper parameters described in Table 3. These parameters are determined empirically according to a series of experiments carried on the whole dataset that give the best results of classification.

As shown in Table 3, the stochastic gradient descent (SGD) algorithm is used in our model to learn the best set of weights and biases of the neural network that minimize the loss function. While learning, the SGD algorithm works by randomly choosing a small number of training inputs. We refer to this as the batch size which is set to 10. The learning rate is set to 0.001. It is the rate at which a function move through the search space. A small learning rate leads to more precise results but it requires more training time. The momentum is an additional factor to determine how fast the SGD algorithm converges on the optimum point. It is set to

Ordination Scatterplot	Map	Scatter plot	Line Chart	Dendrogram	Column Chart	Heat map	Box plot	Area Chart	Pie Chart	Node Link Diagram
279	468	420	412	243	476	214	203	212	245	205

Table 2: Annotated dataset count

Parameter	Optimization algorithm	Learning rate	Momentum	Weight decay	Batch size	Activation function	Iterations
value	SGD	0.001	0.9	0.005	10	Sigmoid	80

Table 3: Hyper parameters choices

0.9.
Also, the effectiveness of our system is evaluated using accuracy, precision, recall and F1-measure. They are defined as follows:

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

$$F1-measure = 2.\frac{P.R}{P+R} \quad (5)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

Where **TN**(True Negative) denotes the case of a negative sample being predicted negative (e.g., a non-line chart image being classified into the complement class of line chart correctly); **TP**(True Positive) refers to the case a positive sample being predicted positive (e.g., a line chart image being classified into the class of line chart correctly); **FN** (False Negative) refers to the case that a positive sample being predicted negative (e.g., a line chart image being classified into the complement class of line chart incorrectly); **FP** (False Positive) denotes the case that a negative sample being predicted positive (e.g., a non- line chart leaf image being classified into the class of line chart incorrectly); **P** refers to precision and **R** refers to recall.

The experimental results of the proposed method are summarized in Table 4 and Figure 2 .

As shown in Table 4, our model has achieved better performance than LeNet which is expected since our model is deeper (has more layers).
Many approaches in the literature return to pre-trained networks to effectively classify their small data. This pre-trained network will learn common useful features from a large dataset such as the ImageNet dataset and then fine-tune it (re-train it) on another dataset with a very small weight updates.

Model	Precision	Accuracy	Recall	F1-measure
LeNet [LeCu 89]	0,885	0,795	0,765	0,821
Pretrained LeNet	0,822	0,409	0,372	0,512
Our Model	0,906	0,895	0,893	0,902

Table 4: Comparison with existing models

However, as can be noticed in Table 4, pre-trained model was not able to provide good results in our case. This could be because of the big differences between natural images contained in the ImageNet dataset and the chart images in our dataset which lead to the low performance of the model.

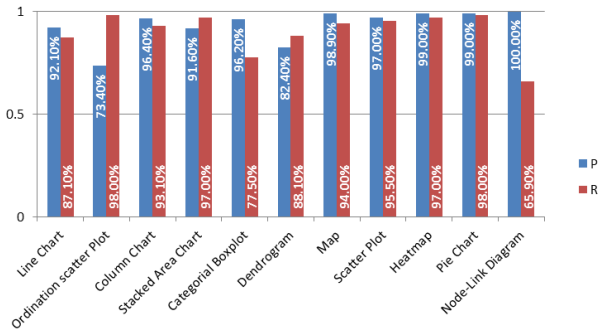


Figure 2: Obtained recall and precision per class

The performance per class is presented in Figure 2. These scores show that our method produces good recall (R) and precision (P) rates.
Our method performs well on heterogenous data in terms of color and monochrome images and considers challenging conditions like orientation and scale. However, we think that the lowest recall rates in all our experiments is due to similarity beteween the visual appeareance of chart images within some classes. This is the case for example of Ordination scatter plot that can easily be confused with Node link Diagram and Scatter plot which consequently affect the recall rate of these three classes. Figure 3 illustrates such similarity in visual appearance.

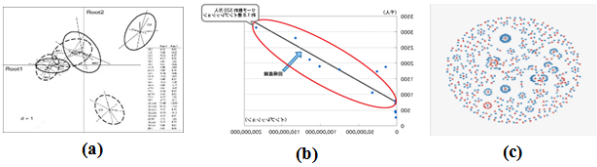


Figure 3: Illustration of confusing chart images (a) ordination scatterplot,(b) Scatterplot, (c) Node link Diagram

Consequently, the confusion in the learning process is mainly caused by the similar visual appearance of images belonging to different classes. In fact, reducing the size of each input image within the step of normalization and subsampling makes it hard to find and extract discriminating features from each image classes.

6 CONCLUSION

The focus of this paper is on the classification of chart images contained in scientific publications especially in the biodiversity research. This work is motivated by the importance of automatically classifying the chart type as an aid in the development of domain knowledge base that can further assist in the visualization recommendation as well as for chart analysis and redesign.

The proposed approach is based on convolution neural networks for chart image classification. We have collected 96837 chart images from scientific publications and have annotated 3377 images according to 11 classes (Categorical Boxplot, Column Chart, Dendrogram, Heatmap, Line Chart, Map, Node-Link diagram, Ordination Scatterplot, Pie Chart, Scatterplot and Stacked Area Chart). Although the challenging conditions of low resolution, dissimilarity, variation in size, color (binary and colored images) and low variance between graphics, the experimental results demonstrate the efficiency of our proposed method.

As future direction, we will focus on the segmentation of each type of chart image and extract informative clues as textual information, shape and area distribution and then provide support for chart redesign and interpretation. Furthermore we will work on multiple type of graphics since they are commonly included in scientific publications.

7 ACKNOWLEDGMENTS

We are grateful to Birgitta König-Ries for her discussions and her technical support. A part of this research was supported by the DAAD funding through the Bio-Dialog project. This work is partly funded by the DFG Priority Program 1374 "Biodiversity-Exploratories" (KO 2209 / 12-2).

8 REFERENCES

[Arel 10] I. Arel, D. C. Rose, and T. P. Karnowski. "Deep machine learning-a new frontier in artificial intelligence research [research frontier]". *IEEE Computational Intelligence Magazine*, Vol. 5, No. 4, pp. 13–18, 2010.

[Deng 14] L. Deng. "A tutorial survey of architectures, algorithms, and applications for deep learning". *APSIPA Transactions on Signal and Information Processing*, Vol. 3, p. e2, 2014.

[Huan 07] W. Huang and C. L. Tan. "A system for understanding imaged infographics and its applications". In: *Proceedings of the 2007 ACM symposium on Document engineering*, pp. 9–18, ACM, 2007.

[Inok 00] A. Inokuchi, T. Washio, and H. Motoda. "An apriori-based algorithm for mining frequent substructures from graph data". In: *European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 13–23, Springer, 2000.

[Kart 12] V. Karthikeyani and S. Nagarajan. "Machine learning classification algorithms to recognize chart types in portable document format (pdf) files". *International Journal of Computer Applications*, Vol. 39, No. 2, 2012.

[LeCu 89] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. "Backpropagation applied to handwritten zip code recognition". *Neural computation*, Vol. 1, No. 4, pp. 541–551, 1989.

[Lee 15] P.-S. Lee and B. Howe. "Dismantling Composite Visualizations in the Scientific Literature". In: *Proceedings of the International Conference on Pattern Recognition Applications and Methods - Volume 2*, pp. 79–91, SCITEPRESS - Science and Technology Publications, Lda, 2015.

[Pras 07] V. S. N. Prasad, B. Siddiquie, J. Golbeck, and L. S. Davis. "Classifying computer generated charts". In: *Content-Based Multimedia Indexing, 2007. CBMI'07. International Workshop on*, pp. 85–92, IEEE, 2007.

[Savv 11] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, and J. Heer. "Revision: Automated classification, analysis and redesign of chart images". In: *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pp. 393–402, ACM, 2011.

[Shao 05] M. Shao and R. P. Futrelle. "Recognition and classification of figures in PDF documents". In: *International Workshop on Graphics Recognition*, pp. 231–242, Springer, 2005.

[Yoko 97] N. Yokokura and T. Watanabe. "Layout-Based Approach for extracting constructive elements of bar-charts". In: *International Workshop on Graphics Recognition*, pp. 163–174, Springer, 1997.

[Zhou 01] Y. Zhou and C. L. Tan. "Learning-based scientific chart recognition". In: *4th IAPR International Workshop on Graphics Recognition, GREC*, pp. 482–492, Citeseer, 2001.

Isomorphic Loss Function for Head Pose Estimation

Iulian Felea, Corneliu Florea, Constantin Vertan, Laura Florea
Image Processing and Analysis Laboratory,
University Politehnica of Bucharest
Spaliul Independentei 313, Bucharest, Romania
{iulian.felea; corneliu.florea; laura.florea; constantin.vertan}@upb.ro

ABSTRACT

Accurate head pose estimation is a key step in many practical applications involving face analysis tasks, such as emotion recognition. We address the problem of head pose estimation in still color images acquired with a standard camera with limited resolution details. To achieve the proposed goal, we make use of the recent advances of Deep Convolutional Neural Networks. As head angles with respect on yaw and pitch are continuous, the problem is one of regression. Typical loss function for regression are based on L_1 and L_2 distances which are notorious for susceptibility to outliers. To address this aspect we introduce an isomorphic transformation which maps the initially infinite space into a closed space compressed at the ends and thus significantly down–weighting the significance of outliers. We have thoroughly evaluated the proposed approach on multiple publicly head pose databases.

Keywords

Head Pose Estimation; Loss function; L_1 distance; Isomorphic transformation.

1 INTRODUCTION

The head pose estimation is the process of inferring the orientation of a human head from digital images. Robust and efficient algorithms for head pose estimation are necessary in a plethora of applications such as human-computer interfaces (where the orientation and head trajectory may be encoding a message by itself, as is the case of nodding), emotion where the facial expression is complemented by head pose (such as for stress) or face recognition analysis. The problem is made more difficult by the variation of the illumination angle and intensity, the variability of the human faces and those of the environment. Another aspect which is lacking somehow is the existence of annotation for large, "in–the–wild" databases.

In computer vision, the estimation of the head pose refers to inference of the orientation of a person's head, relative to the view of a camera. Following the seminal review on the topic by Murphy-Chutorian and Trivedi [MCT09], a more rigorous definition of the head pose estimation is the "ability to infer the orientation of a head relative to a global coordinate system, but this subtle difference requires knowledge of the intrinsic camera parameters to undo the perceptual bias from the perspective distortion". Typically, the head pose is modelled in a three-dimensional space by three Euler angles of rotation around three axis orthogonal to each other: the yaw, the pitch and the roll, as showed in figure 1.

To give an anatomical background let us recall the work of Ferrario et al. [FSS+02] who measure the range of head motion for an average adult male. The following values have been determined: the sagittal flexion and extension (i.e., forward to backward movement of the neck) from -60.4^0 to $+69.6^0$, a frontal lateral bending (i.e., right to left bending of the neck) from -40.9^0 to $+36.3^0$ and a horizontal axial rotation (i.e., right to left rotation of the head) from -79.8^0 to $+75.3^0$.

Due to its importance, many proposals for the head pose problem exists [MCT09]. Yet, the strong majority of methods approach the problem using as input sequences of frames (videos), or by 3D data (i.e. including depth information).

Using only standard image sequence, without depth information, with tracking (and thus working on the frame correlation to diminish the error), one should note the work of Asteriadis et al [ASKK09] which use Distance Vector Fields to locate face features and follow by head pose regression; the same authors continued, [AKK14], by showing that building models oriented per person's, better performance is at hand. Saragih et al. [SLC11] used a constrained local model, with parameter correlated over consecutive frames and improved convergence of the model by mean shift. Valenti et al. [VSG12] showed that combining independently extracted head pose information and eye pupil

(a) (b) (c)

Figure 1: Illustration of the three head rotations: yaw (a - image from Columbia database), the pitch (b) and roll with pitch (as images from HPEG database) .

localization data, can boost the performance in either problem.

In the case of independently considered frames, we note the method proposed by Kumano et al. [KOY+09] who used a variable-intensity template in a Bayesian framework for a combined head pose and face expression analysis. Vincente et al. [VHX+15] Jeni and Cohn [JC16] used various face fiducial point locator to infer the 3D head pose as a preamble for gaze estimation. Preda et al. [PFSF15], relied on HoG and Local Binary Pattern (LBP) to describe input data and on Multi-Layer-Perceptron for regression.

Lately, following the winning of the ILSVRC competition by the Alexnet architecture [KSH12], the dominant solution in various sub–aspects of image classification became the Deep Convolutional Neural Networks (DCNN). Recently, the residual network architecture (ResNet) [HZRS16] have gained the upper hand. The DCNN have been used, previously, in head pose estimation too. For instance, in [MR15] a classifier corrects the indication of a regressor for gathering head pose data as input for human–robot interaction. Recently Venturelli et al. [VBVC17] used RGB-D data from and RGB-Kinect camera to regress, by means of DCNN, the head angles.

Our approach assumes that the head orientation is given in conjunction with faces detected by a specifically trained detector. In the proposed solution the DCNN based detector proposed by Zhang et al. [ZZLQ16] is used. Furthermore, as the authors have released a second version of the algorithm, named Multi-task Cascaded Convolutional Networks -2 (MTCCN-2), which showed improved accuracy, we have adopted too.

The remainder of this paper is organized as follows: Section 2 presents the Deep Neural Network architecture and the adaptation of the loss function proposed; Section 3 summarizes the main features of the databases, while Section 4 presents the achieved results. The paper ends with discussions and proposes further developments.

2 METHOD

2.1 Architecture and Regression Loss

As mentioned in the introductory section, we have relied on DCNN for the actual task of head pose estimation. We have experimented with two standard architectures, namely AlexNet [KSH12] and ResNet50 (Residual network with 50 layers as described in [HZRS16]). In all cases we have used the fine-tuning paradigm: both networks have been previously trained on ImageNet and the found weights initialize the training here. Since we aim to predict head-pose for yaw and pitch, the output dimensions of the last fully connected layer, initially set at 1000 (number of classes in ImageNet), was resized at only 2: one predicts the horizontal angle and the other the vertical angle of the head. Finally, we replaced the last Softmax layer with a custom regression loss layer based on the L_1 distance between ground truth information and the predicted angles.

The loss function is in the simplest case the mean absolute angular error of the head pose estimation for the forward pass:

$$\begin{aligned} \mathscr{L}^{(1)}_{MAE} &= \frac{1}{2N}\sum_{i=1}^{N}\left(|x_i^h - y_i^h| + |x_i^v - y_i^v|\right) \\ &= \frac{1}{2N}\sum_{i=1}^{N}\left(L_1(x_i^h; y_i^h) + L_1(x_i^v; y_i^v)\right) \end{aligned} \tag{1}$$

where N denotes the number of images used by the neural network at training, x_i - the ground truth value (the true horizontal-h/vertical-v angle of the head) and y_i is the estimated angle. $L_1(a,b)$ is the absolute error between a and b, $|a-b|$. The backward pass assumes the derivatives of the $\mathscr{L}$ distance with respect to the given axis.

$$\begin{aligned} \frac{d\mathscr{L}}{dx} &= \sum_{i=1}^{N}\frac{dL_1}{dx_i^h} + \frac{dL_1}{dx_i^v}; \\ \frac{dL_1}{dx_i^k} &= \frac{x_i^k - y_i^k}{|x_i^k - y_i^k|} = sgn(x_i^k - y_i^k). \end{aligned} \tag{2}$$

2.2 Linear label normalization

In regression problems, the training procedure usually optimizes a low value Minkovski distance (typically L_1 or L_2). The loss function is often formed

by adding a regularization term, where the goal is the distance between the estimated values of the network and the ground-truth. However, it is generally known that L_1 and L_2 norm minimization is sensitive to outliers, which can result in poor generalization depending on the amount of outliers present during training [Hub11]. We recall that outliers are rare samples that sometimes appear in the training data, generated, for instance, by facial makeup, rare illumination conditions etc. The outliers can be associated with samples with noisy ground-truth annotation.

In the presence of outliers, the main issue of using L_1 loss in regression problems is that outliers can have a disproportionately high weight and thus, will influence the training procedure by reducing the generalization ability. Previously, the problems was approached by using more elaborated loss functions such as based on Tukey biweight function [BRCN15].

An intuitive solution is to transform the initial set of labels into a more compact one. Under such auspices, eq. (1) becomes:

$$\mathscr{L}^{(2)} = \frac{1}{2N}\sum_{i=1}^{N}\Big(F(L_1(x_i^h;y_i^h)) + F(L_1(x_i^v;y_i^v))\Big) \quad (3)$$

where F is a 1D mapping function that aims to intrinsically reduce the weight of outliers.

Assuming that the given labels range into $[Y_{min};Y_{max}]$ a simple implementation of F is linear mapping into [-1,1]:

$$F(y) = 2\frac{y - Y_{min}}{Y_{max} - Y_{min}} - 1 \quad (4)$$

2.3 Isomorphic mapping

Although simple and intuitive, eq. (4) does not redistribute the weight of the outliers with respect to the weight of normal data. Its positive effect, when exists, can only be attributed to mapping the labels into a domain that is more accessible to a neural networks.

A truly positive mapping would re-weight outliers, which should be in minority with respect to normal data. Assuming that initial data is given into the real number space, and errors close to infinity may exist due network producing completely wrong predictions, one wishes to map them into a closed algebraic space, where predictions have an upper bound. Such a mapping is:

$$F(y) = \frac{y}{1+|y|} \quad (5)$$

given that the input label, $y = y_i$ is in a symmetrical range (i.e. $-Y_{min} = Y_{max}$). The mapping compresses the higher domain, thus under–weighting large errors, which should be in minority when compared to smaller ones that form the majority.

We note that the function defined by eq. (5) is the generative function of the symmetrized version of the logarithmic-like image processing model [VOFF08, NDC13]. In that application the function acts as an isomorphism between $(-1,1)$ range and the image definition range $[0;D_{max} = 255]$. As it is an isomorphism, it also preserves the topology of the initial space [Opp67]. In other words, it means:

$$F(L_1(x;y)) = L_1(F(x);F(y)) \quad (6)$$

The practical advantage is that instead of replacing eq. (5) into eq. (3) and obtaining a more complicate loss function (which implies also more complicated derivatives for the backward pass), one can simply apply the function to all labels and follow by training and testing into $(-1;1)$ space; next one will simply transform the prediction by the inverse of the isomorphic mapping, that is:

$$F^{-1}(x) = \frac{x}{x-1} \quad (7)$$

3 DATABASES

For our experiments we used Boston University Head Tracking [lSA00], Columbia Gaze [SYFN13], Head Pose and Eye Gaze (HPEG) [ASKK09], INRIA [GL04] and UPNA [ABVC16] datasets for head pose estimation. A set of illustrative images from the databases are in Figure 2.

Boston University Head Tracking Dataset. The Boston database [lSA00] contains short video sequences at a resolution of 320×240 pixels with uniform and varying illuminations. Each type of illumination videos was recorded with different subjects. For example, five participants were recorded in 45 videos with normal illumination and only three subjects recorded 27 videos with varying illuminations. To achieve head pose and orientation ground truth information, the subjects wear a magnetic tracker that collected head movement. Each one of them was instructed to perform various head movements, such as rotations, translations.

Columbia Gaze Dataset. The Columbia database [SYFN13] contains 5880 de images of 64 persons with 21 gaze directions and 5 head positions from ethnically diverse subjects. The subjects ranged from 18 to 36 years of age and 21 of them wore prescription glasses. The main purpose of this database is estimation of the point of gaze and thus the images have high-resolution: 5184×3456 pixels. Each of the 64 de persons was recorded with yaw head angle of 0, 15^0 and 30^0 left

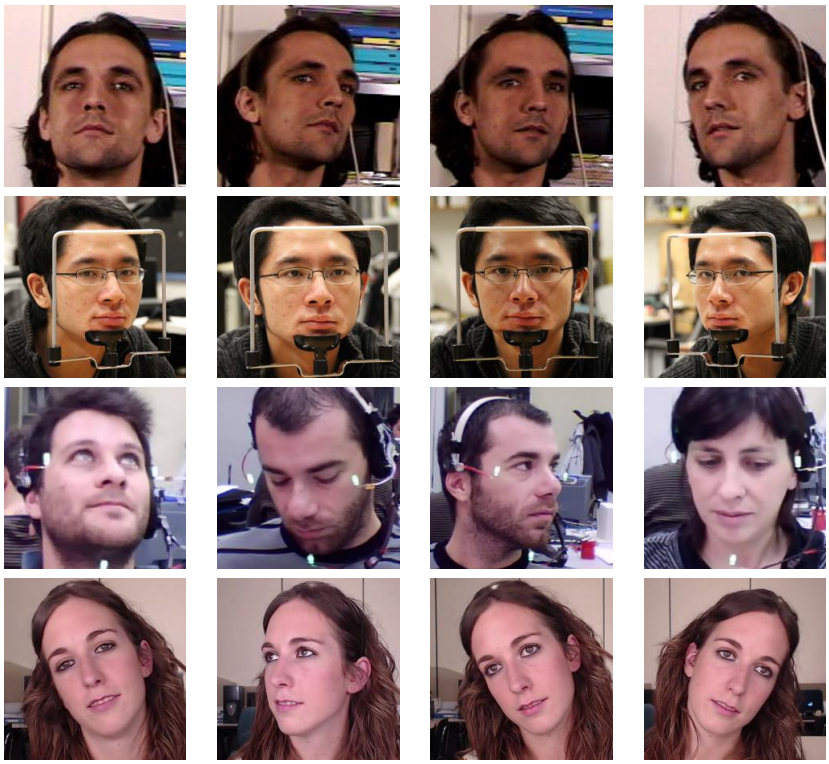


Figure 2: Various head poses from the used databases. Rows from top to bottom: Boston, Columbia, HPEG, UPNA. Note the diversity inside and between databases.

and right. The images were taken in a laboratory and various head angles were obtained by moving the recoding camera, while the subjects stood still.

Head Pose and Eye Gaze (HPEG) Dataset. The HPEG database [ASKK09] is given as videos of 640 × 480 resolution. It is separated in two section differing by the subject to camera distance. It contains data from 10 persons and we extracted all the frames from the given sequences. The head position includes yaw and pitch variations from -30^0 to $+30^0$. The database comes with annotation on head angle obtained from a magnetic sensor placed on top of the subject head.

INRIA Head Pose Dataset The INRIA (also named Pointing04) database [GL04] consists of 15 sets of images acquired with different people. Subjects have between 20 to 40 years old, some have facial hair and seven are wearing glasses. Head pose variation includes yaw and pitch and ranges between -90 and +90 degrees, with a step of 15 degrees for yaw, and 30 and 15 for pitch. Negative values for pitch correspond to bottom poses and positive values correspond to top poses.

UPNA Dataset. The UPNA database [ABVC16] contains a set of 120 videos, acquired from 10 different subjects (6 males and 4 females) of 12 videos each. Every set of 12 videos is composed of six guided-movement sequences and six free-movement sequences. In the guided sequences, the user follows a specific pattern of movement. In the free sequences, the user moves the head at will. In order to provide the database with more uniformity, it has been considered convenient that every video begins and ends with the head in a frontal position, at a working distance from the camera (55–60 cm). Movement ranges include translations going up to more than 200 mm in any axis from the starting point, and rotations up to 30 degrees.

4 IMPLEMENTATION AND RESULTS

4.1 Implementation details

The various alternatives described in this paper were implemented in Matlab. The DCNN architecture support is based on MatConvNet library [VL15].

We ran various tests scenarios. We started with a pretrained AlexNet on ImageNet which was altered as described in section 2.1; for both training and testing there were used Boston, Columbia, HPEG and INRIA databases. The training contains 70% while testing 30%, randomly selected from each of the databases. The results are: $MAE_{Pitch} = 1.00$ and $MAE_{Yaw} = 1.27$. The performance is extremely accurate thus, given the size of the DCNN and the correlation between individual frames, the network memorized the databases.

Given the later observation the train set is formed by joining all images from Columbia, HPEG and INRIA

Table 1: Head pose estimation on Boston dataset when various loss functions and normalization scenarios are envisaged in conjunction with ResNet-50.

Architect.	Normaliz.	Pitch	Yaw
AlexNet	None	4.44	9.53
ResNet-50	None	3.65	9.02
ResNet-50	DB-EQ	2.60	6.32
ResNet-50	DB-EQ + LIN	2.53	5.11
ResNet-50	DB-EQ + ISO	2.32	4.61

while Boston forms the test. In this case the performance dropped to $MAE_{Pitch} = 4.44$ and $MAE_{Yaw} = 9.02$. It shows clearly that generalization is poor over database as there is insufficient variation in data.

Database equalization

At this step we added UPNA database in the training set and still leave Boston completely only in the testing. The problem in this scenario is that UPNA has considerable more images and during training it dominates and the network learns it. Consequently, we balance the contribution of each dataset, by randomly selecting only 6000 images from UPNA.

4.2 Results

The various results, can be seen in Table 1. We tested with both mentioned architectures: AlexNet and ResNet-50. We have experimented with various types of normalization and loss function as follows: no normalization (marked as "None"), database equalization ("DB-EQ") by dropping the contribution of UPNA, linear loss normalization ("LIN") and isomorphic loss normalization marked by ("ISO").

Regarding the results, the first note is that ResNet-50 architecture dominates AlexNet. Database equalization helps, as the system does no learn UPNA any longer. Also the isomorphic loss function shows better performance than linear as, indeed, large errors are in minority.

4.3 Comparison with State of the Art

Multiple methods have reported mean absolute accuracy on the Boston database. None (us included) have trained on Boston but only tested. The comparative results may be followed in table 2. Also on our case, due to the fact that the inverse of of the isomorphism, at the end is susceptible to augment values, sometimes we get prediction far outside the given range. We have bounded these values to range of possible head angles.

The results showed that we obtain the smallest error for the pitch angle variation and the second average error, in the condition that we do not take into consideration any temporal correlation.

Table 2: Comparative performance on the Boston database. Please note that all other methods do head tracking thus reducing the error by enforcing correlation among consecutive frames. In contrast, we report prediction on independent frames, without temporal correlation. With bold letter we have marked the best result for each category.

Method	Pitch	Yaw	Mean
La Cascia et al. [lSA00]	6.1	**3.3**	4.7
Asteriadis et al. [ASKK09]	3.82	4.56	4.7
Kumano et al. [KOY+09]	4.2	7.1	11.3
Valenti et al. [VSG12]	5.26	6.10	5.68
Saragih et al. [SLC11]	4.5	5.2	4.85
Vincente et al. [VHX+15]	6.2	4.3	5.25
Jeni&Cohn [JC16]	2.66	3.93	**3.3**
Proposed	**2.32**	4.61	3.46

5 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a simple method to alter the regression loss so to impose more uniform error so to obtain an accurate head pose angle on individual frames. The algorithm consist in based on Deep Convolutional Neural Networks for both face detection where we have relied on the MTCCN-2 solution and for head pose estimation.

Further developments may envisage domain adaptation between database to be able to relevantly train on various databases and transferred the learned knowledge on the test one. Also the process of adjusting the loss function may assume further optimization with respect to a given error.

6 ACKNOWLEDGMENTS

The authors would like to thank NVidia Corporation for donating the Tesla K40c GPU that helped us run experimental setup for this research. The work is partially financed by University Politehnica of Bucharest,through the "Excellence Research Grants" Program, UPB – GEX. Identifier: UPB–EXCELENTA–2016, Contract no. 95/26.09.2016 (aFAST).

7 REFERENCES

[ABVC16] M. Ariz, J. Bengoechea, A. Villanueva, and R. Cabeza, *A novel 2d/3d database with automatic face annotation for head tracking and pose estimation*, CVIU **148** (2016), 201–210.

[AKK14] S Asteriadis, K Karpouzis, and S Kollias, *Visual focus of attention in non-calibrated environments using gaze estimation*, IJCV **107(3)** (2014), 293–316.

[ASKK09] S. Asteriadis, D. Soufleros, K. Karpouzis, and S. Kollias, *A natural head pose and eye gaze dataset*, ACM Workshop on Affective Interaction in Natural Environments, 2009, pp. 1–4.

[BRCN15] V. Belagiannis, C. Rupprecht, G. Carneiro, and N. Navab, *Robust optimization for deep regression*, ICCV, 2015, pp. 2830–2838.

[FSS+02] V. F. Ferrario, C. Sforza, G. Serrao, G. Grassi, and E. Mossi, *Active range of motion of the head and cervical spine: a three-dimensional investigation in healthy young adults*, J. Orthopaedic Research **20(1)** (2002), 122–129.

[GL04] N. Gourier and J. Letessier, *The pointing 04 data sets*, ICPR, Visual Observation of Deictic Gestures, 2004.

[Hub11] P. Huber, *Robust statistics*, Springer, 2011.

[HZRS16] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, CVPR, 2016.

[JC16] L. A. Jeni and J. F. Cohn, *Person-independent 3d gaze estimation using face frontalization*, CVPR Workshops, 2016, pp. 792–800.

[KOY+09] S. Kumano, K. Otsuka, J. Yamato, E. Maeda, and Y. Sato, *Pose-invariant facial expression recognition using variable-intensity templates*, IJCV **83** (2009), no. 2, 178–194.

[KSH12] A. Krizhevsky, I. Sutskever, and G. Hinton, *Imagenet classification with deep convolutional neural networks*, NIPS (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), 2012, pp. 1097–1105.

[lSA00] M. laCascia, S. Sclaroff, and V. Athitsos, *Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models*, IEEE T. PAMI **22(4)** (2000), 322–336.

[MCT09] E. Murphy-Chutorian and M. Trivedi, *Head pose estimation in computer vision: A survey*, IEEE T. PAMI **31(4)** (2009), 607–626.

[MR15] S. Mukherjee and N. Robertson, *Deep head pose: Gaze-direction estimation in multimodal video*, IEEE Trans. on Multimedia **17** (2015), no. 11, 2094–2107.

[NDC13] Laurent Navarro, Guang Deng, and Guy Courbebaisse, *The symmetric logarithmic image processing model*, Digital Signal Processing **23** (2013), no. 5, 1337–1343.

[Opp67] A. V. Oppenheim, *Generalized superposition*, Information and Control **11** (1967), no. 5,6, 528 – 536.

[PFSF15] V. Preda, C. Florea, A. Sima, and L. Florea, *Simple head pose estimation in still images*, ISSCS), 2015, pp. 1–4.

[SLC11] J. Saragih, S. Lucey, and J. Cohn, *Deformable model fitting by regularized landmark mean-shift*, IJCV **91** (2011), no. 2, 200–215.

[SYFN13] B.A. Smith, Q. Yin, S.K. Feiner, and S.K. Nayar, *Gaze locking: Passive eye contact detection for human–object interaction*, ACM Symposium on User Interface Software and Technology, 2013, pp. 271 – 280.

[VBVC17] M. Venturelli, G. Borghi, R. Vezzani, and R. Cucchiara, *Deep Head Pose Estimation from Depth Data for In-car Automotive Applications*, ArXiv e-prints **1703.01883** (2017).

[VHX+15] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi, *Driver gaze tracking and eyes off the road detection system*, IEEE Trans. on Intelligent Transportation Systems **16** (2015), no. 4, 2014–2027.

[VL15] A. Vedaldi and K. Lenc, *Matconvnet – convolutional neural networks for matlab*, ACM-MM, 2015.

[VOFF08] C. Vertan, A. Oprea, C. Florea, and L. Florea, *A pseudo-logarithmic framework for edge detection*, Advances in Computer Vision (J. Blanc-Talon et al, ed.), LNCS, vol. 5259, Springer Verlag, 2008, pp. 637 – 644.

[VSG12] R. Valenti, N. Sebe, and T. Gevers, *Combining head pose and eye location information for gaze estimation*, IEEE Trans. on Image Processing **21** (2012), no. 2, 802–815.

[ZZLQ16] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, *Joint face detection and alignment using multitask cascaded convolutional networks*, IEEE Signal Processing Letters **23** (2016), no. 10, 1499–1503.

An interactive Tunisian virtual museum through affine reconstruction of gigantic mosaics and antic 3-D models

ELGHOUL,S
CRISTAL Laboratory,
Pole Grift
Campus University
Manouba 2010, Tunisia
sinda.elghoul@ensi-uma.tn

SAIDANI,M
CRISTAL Laboratory,
Pole Grift
Campus University
Manouba 2010, Tunisia
maweheb.saidani@gmail.com

GHORBEL,F
CRISTAL Laboratory,
Pole Grift
Campus University
Manouba 2010, Tunisia
faouzi.ghorbel@ensi.rnu.tn

Abstract

Museums are no longer static depositories for objects, as they used to be for the past two centuries. Online and interactive access has created new opportunities for museums and cultural institutions to reach out and discover new audiences for promotion of cultural achievements. The virtual tours and the panorama reconstructions become among the most popular solutions for virtual consultation of historical monuments. A Tunisian virtual museum application aims to make its relative monuments more accessible to experts such as archaeologist or art historians, and even to the large public. In this research, we intend to present an invariant based approaches for the reconstruction of both mosaic panorama and 3D models. We propose a framework to create a Tunisian virtual museum and we focused on an interactive application related to Bardo museum. We apply affine invariance in a finite set of viewpoints to compute shape descriptors with completeness and stability properties. Such affine invariants serve to refine 3D model generated by classic shape-from-silhouette algorithm and to assist the creation of panorama image mosaics from uncalibrated images.

Keywords

3D, virtual museum, virtual reality, multi-view reconstruction, stitching mosaics.

1 INTRODUCTION

The cultural and historical heritage is consisted of goods that have been created by the previous generations. These objects, because of their symbolic meanings have special value for the current generations and they affect the forming of their identities. UNESCO creates in 1956 ICCROM (International Centre for the Study of the Preservation and Restoration of Cultural Property)[DeCaro2014] offers a real enhancement for conservation of monuments, archeological sites and excavations. Nowadays, virtual museums propose various possibilities to keep that heritage through digitization, analyzing, converting to electronic form and organizing this material. In literature, a virtual museum can be defined as:
a collection of digitally recorded images, sound files, text documents and other data of historical, scientific, or cultural interest that are accessed through electronic media [Malraux1965]

The goal of the virtual museum is not to replace the real museum. Only by cooperation of the ones who work on preserving and presenting the cultural heritage (historians, archeologists, museum curators etc.) can results to cultural creation that can be achieved. Furthermore, we insist that the term of virtual museum can take various forms depending on the application scenario and end-user. It can be a 3D reconstruction of the physical museum or even a completely imaginary environment. In the other hand, with the help of new technologies, V-museums collections [Swartout2010], in place of material ones, present some interesting features: money and space savings, easier management potential, digital analysis or modeling [Pearce1993], etc.

In Tunisia, over 23 sites are registered in UNESCCO as world heritage; however, much of the information produced by archaeological research over the past century exists in technical, sometimes limited distribution reports scattered in offices across the country. Moreover, several museums are spread out on the Tunisian territory: it contains many type of objects related to different historical periods: romain, punic, bysantic etc.

Therefore, the creation of Tunisian virtual museum is today an expected choice to preserve cultural heritage and to help visitors to obtain efficiently any information about the Tunisian archeology without having to visit physically museums. The remainder of this paper will be organized as follows. The section 2 is about state of art of virtual museums, in section 3 we give our methodology for realization of Tunisian virtual museum prototype including panorama images and 3D reconstruction of models, next in section 4 we provide experimental results and finally we conclude with a summary and discussion.

2 VIRTUAL MUSEUM: STATE OF ARTS

In 1947, Andre Malraux introduced the first virtual museum, where he proposed the concept of an imaginary museum [Malraux1965] without walls, with its information surrounding the objects, might be made accessible across the world. From this date, this definition knows a different side of points. Its about three categories that are developed like an extension of physical museums [Hu2016] : 1) the first one is the brochure museum which contains basic information such as resume history, opening hours and sometimes a calendar of events etc. the second one is known as the content museum which is almost a web site containing some information about the museum. The third one is called the learning museum propose many points of access for its visitors, depending on their different age, background, and knowledge [Hu2016]. The main goal is to motivate the virtual visitor to discover more about a particular subject and come back to establish a personal relation with the online collection. Today, museums are committed to the construction of V-Museums as an important international window for exposing different heritages and cultures worldwide [Styliani2009]. In the interest of convergence between the world of culture and new technologies, the most cultural institutions aim to spread and share the treasures of cultural heritage in a virtual way. Archaeological sites, sculptures, masterpieces, monuments, etc. are digitized in high definition to be exhibited via websites or cultural applications allowing users all over the world to discover history and art [Addison2000].Many studies tell that most of the people that try a virtual tour or assist to a virtual exhibition are interested to see physically monuments and archaeological sites. Moreover, some virtual exhibitions offer access to places and museum areas that are usually inaccessible to the public, so users can discover details that are imperceptible to the naked eye. Creating a virtual museum need the collaboration of various methods and technologies like 3D modeling, 360 panoramic photos, digitization in gigapixel format (Google), 3D videos, augmented reality, etc [Styliani2009].
Many monuments have their virtual tour: Giza 3D is a great web application that provides a free 3D tours and permits online visitors to virtually discover Egypt pyramids. This work is a result of the partnership between the Museum of Fine Arts in Boston, Harvard University, and Dassault System. The project used QuickTime Virtual Reality technology. Its contain 1,300 panoramic views inside the program that make users have 360-degree views inside and outside the tombs [der2013]. Beylerbeyi Palace virtual tours was done in the years 2007 and 2008. Panorama shootings were taken by a Nikon D80 digital camera, 10.5 mm fisheye lens, panoramic tripod head, and a tripod. This Turkish Palace contains 672+ pictures [gur2013]. These digital technologies can offer historians and the general public to discover the history and cultural heritage of a site like Versailles 3D offered by Google art project [perriault1997].To visit a famous monument in Paris without being carried away by the crowd and without having to privatize a part to feel sovereign the time of a moment the virtual visit allows to see with extreme precision the paintings and other historical beauties that it would be difficult to observe this by going to a palace [riedinger2015], it uses the street view technology of Google and offers the discovery of the gardens, the King's and Queen's large apartments and the Hall of Mirrors.

3 METHODOLOGY

Here, we propose a framework to create a Tunisian virtual museum. Figure 1, present the global architecture with different steps. In this paper we will focus on creating panorama images for mosaics and 3D reconstruction of objects.

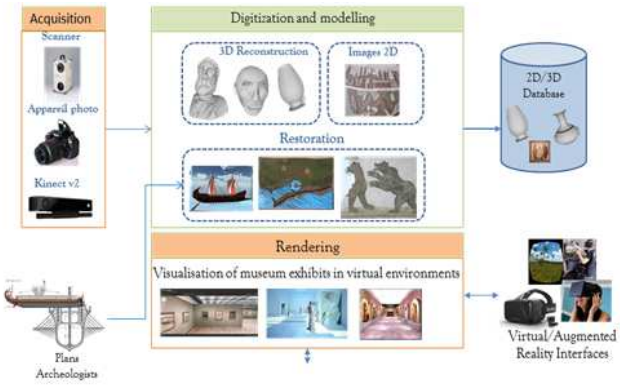


Figure 1: The global architecture of the Tunisian virtual museum.

3.1 Affine acquisition

The acquisition is a key step in virtual museum application. It consists of acquiring high-resolution images for sites and models. In practice, we used CANON

EOS 5D camera (see table 1 for the characteristics) for the different acquisition. So to perform numerical calculations, an algebraic representation is required. thats camera models has been proposed: The most general one is perspective one. Geometric entities: points, lines and planes are represented using homogenous coordinates. In this representation, a point in three-dimensional space is represented by four coordinates which are defined up to a scale factor.

Camera	Canon EOS 5D MarkII
Size of sensor	21 megapixels
Sensitivity	ISO100
Focal Length	24mm
Opening of diaphragm	for f/18 to f/22

Table 1: Technical characteristics of camera

When the distance between object and a camera is much greater than the depth variation affine approximation can be used in stead of respective model. Affine camera, was introduced by [mundy1992], which is widely adopted to simplicity the estimation of the geometry of the acquisition system. weak-perspective or a first order perspective approximation of full An affine camera is a camera whose projection matrix is an affine transformation has the following form:

$$P \sim \begin{pmatrix} P_{2*3} & P_2 \\ 0_3^T & 1 \end{pmatrix} \quad (1)$$

P is an affine matrix projection of a camera whose projection matrix is assumed to affine transformation.

$$P \sim \begin{pmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ 0 & 0 & 0 & P_{34} \end{pmatrix} \quad (2)$$

The last row of the projection matrix is of the form [0, 0, 0, 1] and the matrix have only eight degree of freedom.

The affine camera matrix can be decomposed in: 1. A 3D affine transformation between world and camera coordinate system 2. Parallel projection to onto the image plane 3. A 2D affine transformation

3.2 Affine Mosaic panorama

Bardo Museum contain the second largest collection of ancient mosaics in the world. Some of these large mosaics completely cover the walls of his rooms that have very high ceilings in the order of 10 to 16 meters while the widths of these rooms are both similar. Also, the mosaics are exposed in a museum with laying conditions of the low recoil of the camera compared to the extent of the scene. This state of affairs is made in most cases unable to acquire a mosaic entirely with a single capture. Our goal is to develops an interactive system which integrates high-resolution panoramas and a high color quality of the entire collection of ancient mosaics of the Roman and Byzantine.The Key idea is to take a lot of pictures from multi point of vue and assembled them to find a composite image with a much larger field of view relatively to the size of the acquisition system. Therefore, these acquisitions of piecewise of images obtained in varying positions by cameras undergo a strong apparent deformation. These deformations are known in literature by an affine transformation of the image plane. Also, the variabilities of the illuminates in the same mosaic represent a strong constraint for the construction of panorama[van1992].

In this case of a museum the major defiance are: 1. transformations type like perspective distortions: The deformations transforming the rectangular edge in a kind of trapezium having a large base of reports on a small base increasingly important. 2. Highly variable chrominances gradually and unevenly changing the bottom to the top of the mosaic left and to the right of the camera The image stitching technique can be classified into two approaches: the first one is the direct techniques that compare the intensities of all pixels of the images with each other, while the second based feature techniques are designed to determine a matching between the images with different descriptors extracted from the processed images. Figure 2. These transformations belong to a Lie group noted SA (2, R)) [adel2014].

Figure 2: The proposed stitching pipeline.

Iwasawa decomposition lets say that an element of this, the group can be seen as a movement of the plane followed by stretching[ying2011].

Thus this type of transformations can be considered as a first approximation prospects even if these processes show the case where the scene is located sufficiently far from the camera. In this case, the affine invariant is the best solution.

This is why we propose to study different extractors of keys points that are invariant to the displacements of the plan and those with invariance under affinities[yu2011] such as SIFT(Scale-invariant feature transform) [lowe2004] , MSER (Maximally stable extremal regions) [matas2004] ,color-SIFT (CSIFT) [abdel2006] ,Harris-affine , Hessian-affine [mikolajczyk2004] and Affine-SIFT (ASIFT) [Morel2009] . The descriptors Affine Scale Invariant Feature Transformation (ASIFT) which are known by their invariance with respect to affine transformations are obtained according to pro-

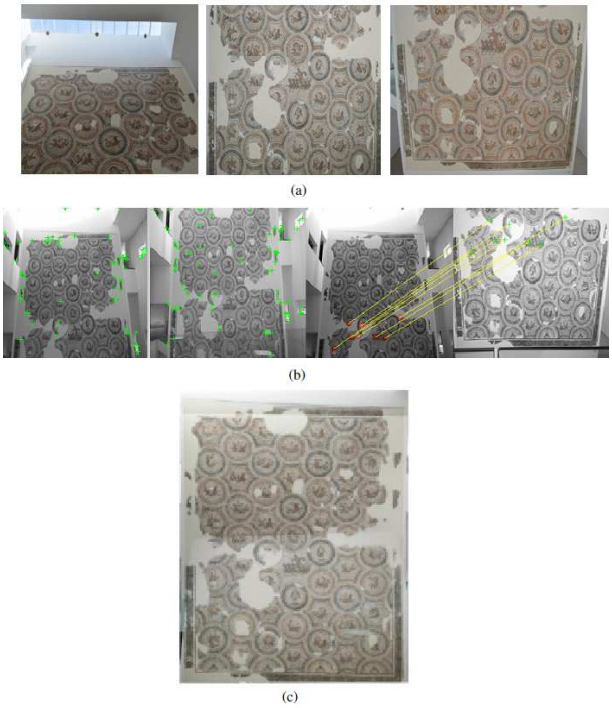


Figure 3: (a) Different acquisitions of the same large mosaic in three shot. (b) ASIFT keys points detection and matching. (c) reconstruction of the mosaic by the proposed stitching algorithm.

cedures similar to those of SIFT [yu2011],[juan2009] , [wu2013] .

They also provide a set of points of interest around which invariant attributes relative affinities are defined. In part of matching, we follow the usual approach, using a k-nearest neighbor algorithm (KNN) to link the keys points and RANSAC (random sample consensus) presented by Martin A. Fischler et al. in [9] to eliminate mismatches, in different variations descriptors.

3.3 Affine 3D reconstruction

Our goal is to reconstruct 3D objects from a sequence of geometrically calibrated images. For this aim, we dispose of several types of information contained in images. Among all the information available, silhouettes are the most useful for shape retrieval.

The reconstruction algorithm, here presented, consists of two phases. In the first phase it computes visual hull of the object using classical carving method [lazebnik2007],[laurentini1994] utilizing camera calibration parameters [esteban2004]. The second phase refines this coarse model using a geometric approach based on aligning contours of real silhouettes and re-projected visual hull silhouettes using affine invariant matching [saidani2012]. Such matching is based on estimating affine transformation between real and re-projected contours. Let O_1 and $O_1^{'}$ be two closed curves that define the same silhouette. O1 and $O_1^{'}$ are said to be related by an affine transformation if and only if:

$$h(l) = \alpha A f(l + l_0) + B \tag{3}$$

Where B is a translation vector, A is linear transformation, l_0 is a shift value, α is a scale factor, f and h are their respective re-parametrization by affine arc length. Such re-parametrization is affine invariant [ghorbel1998]. In Fourier space we get:

$$U_K(h) = \alpha e^{2i\pi k l_0} A U_K(f) + B\delta_k \tag{4}$$

Where $U_K(f)$ and $U_K(h)$ are respectively the Fourier coefficients of f and h. So we could estimate the affine motion between a pair of contours by estimating affine motion parameters: the affine matrix A, the shift value l_0, and the scale factor α. For more details we refer to [saidani2012].

For the proposed refinement algorithm (algorithm 1), we propose to compute a signed distance to the visual hull. For a P_M point, we determine the minimal negative Euclidean distance d(VH,P_M) between real position of P_M in silhouettes and the projection of this point on new silhouettes.

$$d(VH, P_M) = mind(S_i, P_i P_M) \tag{5}$$

Where P_i denotes the projection matrix of the camera i and S_i is the i^{th} image silhouette. Negative distance implies that the contour projection is outside silhouette, so we propose to iteratively perform surface reconstruction by re-carving visual hull according to the affine contour alignment until this distance become positive. For more details concerning results of 3D reconstruction the reader can be referred to [saidani2016].

4 TOWARD A TUNISIAN VIRTUAL MUSEUM: EXPERIMENTAL RESULTS

The goal of this virtual museum is to connect Tunisian archaeological sites and objects of the whole country in order to emphasize its importance as well as to point out the need for preserving the heritage and to open the possibility for future international cooperation between Tunisia and Mediterranean countries.

For this demonstration, we choose to reconstruct Bardo museum. The Bardo is one of the most important museums of the Mediterranean basin. It traces the history of Tunisia over several millennia and through many civilizations through a wide variety of archaeological pieces. It contains the largest collection of Roman mosaics in the world and other antiquities of interest from

Figure 4: Real acquired sequence of archeological objects: 36 calibrated frames of "hannibal" and "punic mask" and their correspondent silhouette sequence.

Algorithm 1 Visual hull refinement based affine invariance

Require: projection matrix P_i, original contours sequences C_i and reprojected contours sequence $C_i^{'}$,$i \in [1..36]$ compute N the affine normalization on contour C_i and $C_i^{'}$
for all C_i **do**
repeat
Compute (α, l_0, A) the affine transformation parameters between C_i and $C_i^{'}$
Compute $D = mind(C_i, C_i^{'})$
Increase VH resolution ocTree algorithm.
Project Ray $l = P_i * N$
Compute the 2D interaction interval $I = l \bigcap C_i$
Re-cave visual hull: $VH^{'} = VH \bigcap P_i^{-1} \bigcap I$
until $(D \geq 0)$
end for

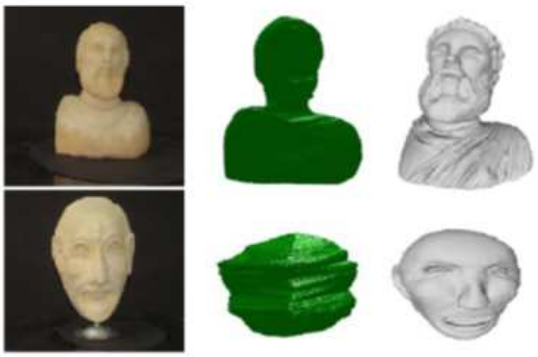


Figure 5: 3D reconstruction of a real archeological object.(from left to right) image extracted from the acquired sequence, the corresponding visual hull and the 3D model.

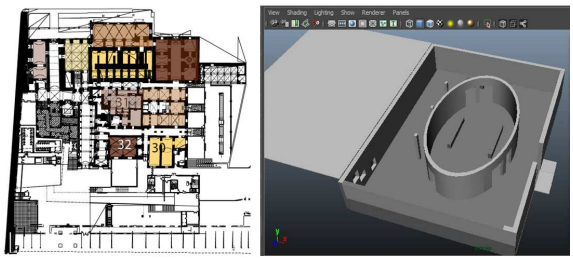


Figure 6: A 2d plan of Bardo museum and its 3D reconstruction of the Bardo.

Ancient Greece, Carthage, Tunisia, and the Islamic period.

The first step in our pipeline is 3D modeling of the Bardo museum environment from initial 2D plans. The results are shown in figure 6.

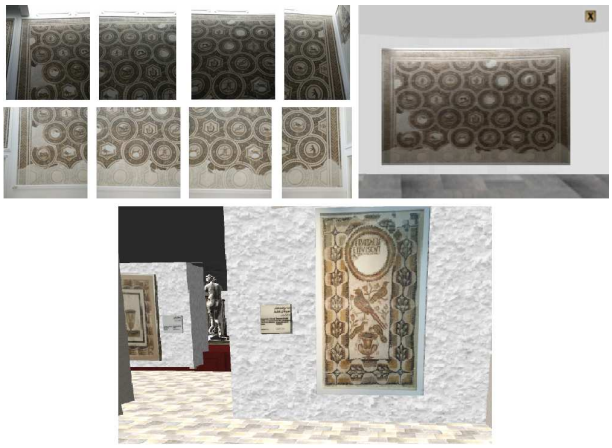


Figure 7: Panoramic image of huge mosaic in virtual environment of Bardo Museum.

Next step of our work consists on integrating panorama images of mosaics and 3D reconstruction of objects results in an interactive application of virtual tour in Bardo museum. Objects are placed in the same way that we retrieve on real visit (figure 7, figure 8). but we provide detailed and up close images of museum artefacts and text descriptions, sometimes with even more information than in the museum itself (figure 9).

5 CONCLUSION

Museums audiences, often regarded as passive participants, now look for interaction and seek to participate in new ways and to have greater access to stored objects. Tunisian virtual museum aims to offer new opportunities for museums and cultural institutions to reach out and discover new audiences and for promotion of cultural achievements in Tunisia. In this paper, we have presented two keys of this framework: panorama mosaics and 3D reconstruction and their integration in a

Figure 8: 3D objects in virtual environment of Bardo Museum.

Figure 9: Historical description of objects in virtual tours.

virtual tour for the Bardo museum. We inderline that the Tunisian virtual museum is a development platform with a possibility of constant input of new data and content, the mapping of new sites and expansion of the database.

6 REFERENCES

[DeCaro2014] De Caro, Stefano . International Centre for the Study of the Preservation and Restoration of Cultural Property (ICCROM)(Conservation and Preservation). Encyclopedia of Global Archaeology,Springer, pp.3937–3940, 2014.

[Pearce1993] Pearce, Susan M and Pearce, Susan M and für Museumsstudien, Professorin. Museums, objects, and collections: A cultural study,Smithsonian Institution Press Washington, DC pp.453-469,1993.

[Swartout2010] Swartout, William and Traum, David and Artstein, Ron and Noren, Dan and Debevec, Paul and Bronnenkant, Kerry and Williams, Josh and Leuski, Anton and Narayanan, Shrikanth and Piepol, Diane and others.Ada and Grace: Toward realistic and engaging virtual museum guides.International Conference on Intelligent Virtual Agents.International Conference on Intelligent Virtual Agents.pp 286–300.2010

[Morel2009] ASIFT: A new framework for fully affine invariant image comparison.Morel, Jean-Michel and Yu, Guoshen.SIAM Journal on Imaging Sciences.Vol 2 pp.438–469.2009

[Malraux1965] Malraux, Andre.Le musee imaginaire ,Gallimard Paris.1965.

[Hu2016] Hu, Qingwu and Yu, Dengbo and Wang, Shaohua and Fu, Caiwu and Ai, Mingyao and Wang, Wende. Hybrid three-dimensional representation based on panoramic images and three-dimensional models for a virtual museum: Data collection, model, and visualization,Information Visualization.2016

[Styliani2009] Styliani, Sylaiou and Fotis, Liarokapis and Kostas, Kotsakis and Petros, Patias,Journal of cultural Heritage vol 10.pp520–528,2009.

[Addison2000] Addison, Alonzo C, Emerging trends in virtual heritage.IEEE multimedia vol 7.pp 22–25.2000

[der2013] Der Manuelian, Peter, Giza 3D: Digital archaeology and scholarly access to the Giza Pyramids: The Giza Project at Harvard University.Digital Heritage International Congress (DigitalHeritage), vol 2.pp 727–734,2013

[gur2013] Gur, Faik,Sculpting the nation in early republican Turkey,Historical Research vol86. pp 342–372.2013.

[perriault1997] PERRIAULT, I, Beatrix Saule: visite en 3D du chateau de Versailles.Archimag,pp 23–35.1997.

[riedinger2015] Riedinger, Christophe and Jordan, Michel and Tabia, Hedi.Restitution 3D de monuments historiques a partir de plans anciens. 2015.

[van1992] Van Gool, Luc J and Brill, Michael H and Barrett, Eamon B and Moons, Theo and Pauwels, Eric.Semi-differential invariants for nonplanar curves.Geometric invariance in computer vision pp 293–309.1992.

[adel2014] Adel, Ebtsam and Elmogy, Mohammed and Elbakry, Hazem,Image stitching based on feature extraction techniques: a survey,International Journal of Computer Applications (0975-8887).2014.

[ying2011] Ying, Shihui and Peng, Yaxin and Wen, Zhijie.Iwasawa decomposition: a new approach to 2D affine registration problem.Pattern Analysis and Applications vol 14.pp127–137.2011.

[abdel2006] Abdel-Hakim, Alaa E and Farag, Aly A. CSIFT: A SIFT descriptor with color invariant characteristics.2006 IEEE Computer Soci-

ety Conference on Computer Vision and Pattern Recognition (CVPR'06) vol 2.pp 1978–1983.

[lowe2004] Lowe, David G.Distinctive image features from scale-invariant keypoints vol 2.pp 91–110.2004.

[matas2004] Matas, Jiri and Chum, Ondrej and Urban, Martin and Pajdla, Tomás,Robust wide-baseline stereo from maximally stable extremal regions.Image and vision computing vol 22.pp 761–767 2004.

[mikolajczyk2004] Mikolajczyk, Krystian and Schmid, Cordelia, Scale & affine invariant interest point detectors,International journal of computer vision vol 60 . pp 63–86 .2004.

[yu2011] Yu, Guoshen and Morel, Jean-Michel,Asift: An algorithm for fully affine invariant comparison,Image Processing On Line,2011.

[wu2013] Wu, Jian and Cui, Zhiming and Sheng, Victor S and Zhao, Pengpeng and Su, Dongliang and Gong, Shengrong. A Comparative Study of SIFT and its Variants.Measurement Science Review vol 13.pp 122–131,2013.

[juan2009] Juan, Luo and Gwun, Oubong,A comparison of sift, pca-sift and surf,International Journal of Image Processing (IJIP) vol 3,2009.

[lazebnik2007] Lazebnik, Svetlana and Furukawa, Yasutaka and Ponce, Jean.Projective visual hulls,International Journal of Computer Vision vol 74 ,pp 137–165,2007.

[laurentini1994] Laurentini, Aldo,The visual hull concept for silhouette-based image understanding,IEEE Transactions on pattern analysis and machine intelligence, vol 16,pp 150–162, 1994

[esteban2004] Esteban, Carlos Hernández and Schmitt, Francis,Silhouette and stereo fusion for 3D object modeling,Computer Vision and Image Understanding vol 96,pp 367–392, 2004.

[saidani2012] Saidani, Maweheb and Ghorbel, Faouzi,Affine invariant descriptors for 3D reconstruction of small archaeological objects,CompIMAGE 2011 pp383–386,2012

[ghorbel1998] Ghorbel, Faouzi, Towards a unitary formulation for invariant image description: application to image coding,Annales des telecommunications vol 53. pp 242–260, 1998.

[mundy1992] Mundy, Joseph L and Zisserman, Andrew, Appendix-projective geometry for machine vision,Citeseer 1992.

[saidani2016] Maweheb, S., Malek, S., and Faouzi, G. (2016). Geometric invariance in digital imaging for the preservation of cultural heritage in Tunisia. Digital Applications in Archaeology and Cultural Heritage, 3(4), 99-107.2016.

Scene text segmentation based on redundant representation of character candidates

Matko Saric
University of Split
Rudjera Boskovica 32
21000, Split, Croatia
msaric@fesb.hr

ABSTRACT

Text segmentation is important step in extraction of textual information from natural scene images. This paper proposes novel method for generation of character candidate regions based on redundant representation of subpaths in extremal regions (ER) tree. These subpaths are constructed using area variation and pruned using their length: each sufficiently long subpath is character candidate which is represented by subset of regions contained in the subpath. Mean SVM probability score of regions in subset is used to filter out non character components. Proposed approach for character candidates generation is followed by character grouping and restoration steps. Experimental results obtained on the ICDAR 2013 dataset shows that the proposed text segmentation method obtains second highest precision and competitive recall rate.

Keywords

Scene text segmentation, extremal regions, SVM classification

1 INTRODUCTION

Extraction of textual information from scene images is challenging problem that is essential for range of different applications like text translation, content based image indexing, reading aid systems for visually impaired people etc. In contrast to the text in document images, recognition of natural scene text is more complex task. This is primarily caused by imaging conditions (perspective distortion, uneven illumination, shadows, blur, sensor noise) and specific properties of scene text (complex background, variable color, font line orientation). Extraction of textual information from scene images consists of three steps:localization, segmentation and recognition [Kar15]. Text localization methods can be classified into 3 groups: sliding-window based methods, methods based on connected components (CCs) and hybrid methods. Methods in first group [Jad14], [Wan12] use window that moves across the image and detect characters using local features. This approach is robust to background noise, but main drawback is computational complexity caused by high number of patches needed to cover text with different scales, aspect ratios and orientations. These methods also don't segment characters from background. Techniques in second group assume that characters are connected components. In this way text segmentation is also provided what can be exploited in recognition step. Maximally Stable Extremal Regions (MSER) region detector is widely used in literature achieving promising performance. Its main drawback is high number of repeating components that affects character grouping algorithm. In [Yin14] character candidates are extracted using MSER segmentation. Pruning of non-character CCs is performed by minimizing regularized variations and exploiting parent-children relations in MSER tree. Character candidates grouping is performed using single-link clustering algorithm with trained distance function. Neumann and Matas [Neu15] proposed extremal regions (ERs) based method for real time localization and recognition. Probability of each ER being a character is determined through two stage classification. First stage employs Adaboost classifier with incrementally computed descriptors, while in second stage remaining candidates are classified using SVM and more complex features. ERs are grouped into lines with highly efficient clustering algorithm followed by OCR module. Sung et al. [Sun15] presented text detection algorithm where firstly ER tree subpaths are constructed using similarity between parent and child regions. Each sufficiently long subpath correspond to characters. Finally, region with lowest normalized variation in subpath is chosen as character candidate. Hybrid text localization methods combines

sliding-window and connected components based approaches. Huang et al. [Hua14] introduced text detection method combining MSER and convolutional neural network (CNN) classifier. MSER segmentation reduces the number of search windows, while sliding-window CNN classifier handles characters merged or missed by MSER. Lu et al. [Lu15] propose scene text extraction method where three text-specific edge features are introduced to detect candidate text boundaries. Character candidates are segmented based on Niblack thresholding of areas found in previous step. Support vector regression (SVR) and bag-of-words (BoW) model are employed to confirm true words after text-line construction. Text segmentation method presented in [Mil15] includes Niblack binarization, calculation of initial labeling confidence using Laplacian of pixel intensities and global optimization. Combined with of-the-shelf OCR software proposed method gives performance comparable to state-of-the-art text reading methods. In this paper novel method for text segmentation is proposed. Main contribution is character candidates generation method based on redundant representation of character candidates. ER tree subpaths are extracted based on area variation. Each sufficiently long subpath is character candidate which is represented by subset of subpath regions. SVM classification is performed on each ER in this subset and mean probability score of the subpath is calculated in order to accept or reject character candidate. In this way redundant representation of character candidates and multiple classifications are exploited to compensate SVM classification errors and obtain more relevant estimation of character probability. Compared to recent work, proposed approach achieves second highest precision and competitive recall for scene text segmentation step.

2 PROPOSED METHOD

An illustration of proposed system is showed in Figure 1. First step is extraction of lightness component from the input RGB image. Next step is ER-based character candidates generation. After that classification of character candidate regions is performed using geometrical features thresholding and SVM classifier. This is followed by character grouping and restoration steps which aims to correct errors from classification step.

2.1 Character candidates generation and classification

Drawback of MSER-based character extraction is high number of repeating components. It should also be noted that maximum stability of the region doesn't guarantee correct character extraction. In order to address these problems a novel method for character candidates selection is proposed in this paper. Firstly, ER tree subpaths are extracted based on ER variation values. Each sufficiently long subpath is considered as

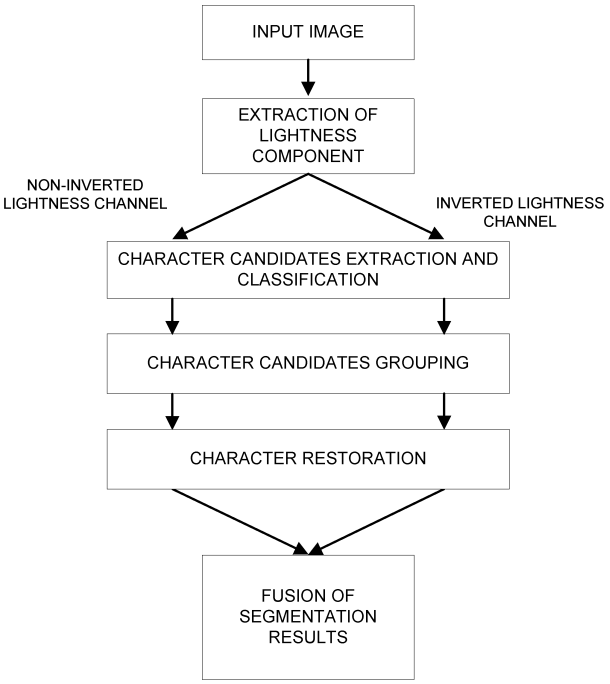


Figure 1: Overview of the proposed method

character candidate. Extremal region [Mat04] is connected component with pixels having lower or higher values than boundary pixels. Binary threshold image $B(p)$ is obtained from gray level image I as

$$B(p) = \begin{cases} 1, & \text{if } I(p) > t \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

where t is threshold and p is pixel position in image I. An extremal region R_t at threshold t is defined with

$$\forall p \in R_t, \forall q \in boundary(R_t) \Rightarrow B_t(p) \geq B_t(q) \quad (2)$$

where p and q are pixel positions. The variation of extremal region R_t is defined as

$$\Psi(R_t) = \frac{(|R_{t+\Delta}| - |R_t|)}{|R_t|} \quad (3)$$

where $|R_t|$ denotes number of pixels in region R_t and Δ is a parameter. Relation between extremal regions is usually represented by ER tree. It is component tree where each node represents extremal region. Edge represents inclusion relation between parent and child region. Parent region R_i and child region R_j satisfy next condition:

$$\bigvee p \in R_i \rightarrow p \in R_j \quad (4)$$

By moving to the root node, value of threshold t decreases what results with larger regions (Figure 2). The root is region covering the whole image. In this paper

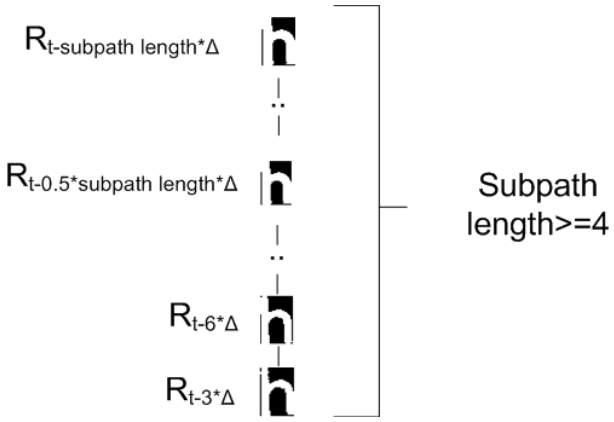


Figure 2: Characters candidate subpath represented by subset of ERs

ER tree subpaths are constructed using variation values. ERs R_t and its child $R_{t+\Delta}$ are split into two subpaths if

$$|\Psi(R_t) - \Psi(R_{t+\Delta})| \geq t_{variation} \quad (5)$$

where $t_{variation}$ is variation threshold that is set to value 0.5 according to [Yin14] and Δ is set to value 4. In this way only subset of regions R_t with $t = k * \Delta, k \in [0, 255/\Delta]$ is considered. Characters correspond to extremal regions with lower variations what is manifested in longer ER subpath.

Therefore, subpath $S = \{R_{t-\Delta}, R_{t-2*\Delta}, ..., R_{t-slen*\Delta}\}$, where *slen* denotes subpath length, is considered as character candidate if $slen >= 4$. Each subpath is represented with subset of regions: $R_{t-j*\Delta}$, where $j = 3*n, n \in [1, subpathlength/3]$ (Figure 2). In this way subpath regions are subsampled by factor 3 to lower the number of classifications needed for one subpath. Regions from this subset are further sent to classification step in order to calculate their SVM character probability scores. Final score is calculated as mean of these probabilities:

$$character\ probability = mean(p_{t-j*\Delta}) \quad (6)$$

where $p_{t-j*\Delta}$ represents SVM probability score of region $R_{t-j*\Delta}$. If $character\ probability \geq 0.5$, subpath is accepted as valid character and in the further steps it is represented by region $R_{t-0.5*subpathlength*\Delta}$ which corresponds to the middle of the subpath. This choice is motivated by fact that regions on the subpath bottom tend to be eroded version of character, while top of the subpath often contains dilated components. Proposed approach exploits redundant subpath representation taking mean of multiple probability scores of the same character candidate in order to compensate SVM classifier errors.

Classification into character and non-character ERs is performed in two steps. Obviously non character objects are eliminated by thresholding the values of area, height, aspect ratio and number of holes. These regions are discarded from the processing pipeline and can't be recovered in character restoration step. Further, SVM classifier is employed to distinguish character ERs from non-character ERs. It was trained on training set consisting of 3198 positive and 4558 negative samples. Positive samples are extracted from ICDAR 2011 training [Sha11], KAIST [Kai17] and Char74k [Cam09] datasets, while negative samples are generated from ICDAR 2011 training and BSD 300 [Mar01] datasets followed by extracting non-character CCs. RBF kernel was used with $\gamma = 0.69$. Following features used in [Gon13] and [Neu15] forms input vector for SVM classifier:

Aspect ratio

$$Aspect\ ratio = \frac{width}{height} \quad (7)$$

where *width* and *height* denote width and height of region bounding box.

Occupy rate

$$Occupy\ rate = \frac{area}{height * width} \quad (8)$$

where *area* is region area.

Compactness

$$Compactness(CC) = area/perimeter^2 \quad (9)$$

where *perimeter* is number of contour pixels.

Solidity (*convex area*):

$$Solidity(CC) = \frac{area}{convex\ area} \quad (10)$$

Occupy rate convex area:

$$Occupy\ rate\ convex\ area = \frac{convex\ area}{height * width} \quad (11)$$

Skeleton length to perimeter ratio

$$Skeleton\ length\ to\ perimeter\ ratio = \frac{skeleton\ length}{perimeter} \quad (12)$$

Stroke width features

$$Mean\ stroke\ width\ size\ ratio = \frac{mean(stroke\ width)}{max(height, width)} \quad (13)$$

$$Max\ stroke\ width\ size\ ratio = \frac{max(stroke\ width)}{max(height, width)} \quad (14)$$

$$Stroke\ width\ variance = \frac{\sigma^2_{stroke\ width}}{mean(stroke\ width} \quad (15)$$

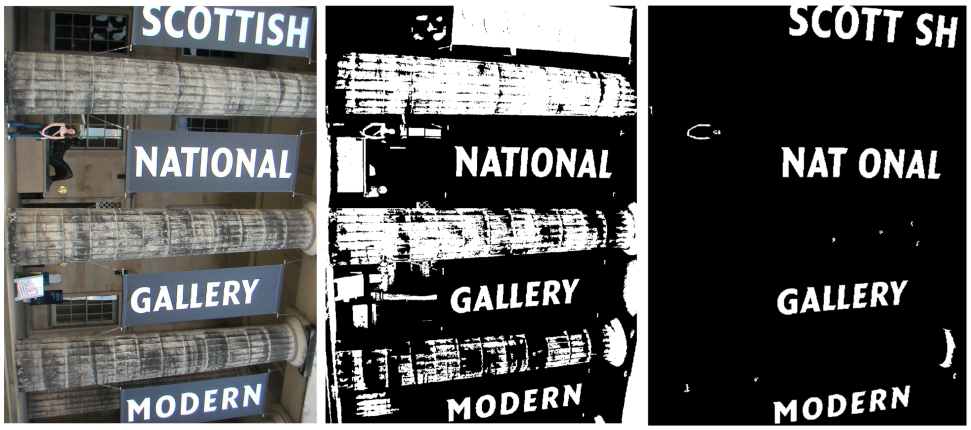


Figure 3: Characters candidates classification: input image (left), generated character candidates (middle) and character candidates after SVM classification (right)

Filled area ratio

$$Filled\ area\ ratio = \frac{abs(Filled\ area - Area)}{Area} \quad (16)$$

where *Filled area* is the number of pixels in region with all holes filled.

Number of inflection points is used as indicator of shape complexity. **Euler number** is the difference between the number of connected components and number of holes.

Regions with SVM probability score lower than 0.5 are temporarily discarded, but they are considered as candidates for character restoration in the last step. Example of character candidate classification is shown on Figure 3. In case of low aspect ratio ERs classifier can easily misinterpret letters like "i" or "l" as noise and vice versa. Therefore, these regions are simply discarded when its aspect ratio is lower than 0.3 assuming that low aspect ratio characters will be recovered in character restoration step.

2.2 Character candidates grouping and restoration

Grouping of character candidates and restoration of missed characters are performed with same method as in [Sar16]. Firstly, extremal regions classified as characters are grouped into text lines. In this step text line properties are exploited to filter out false positive classifications. Procedure consists of the following steps:
1. Regions are firstly sorted by centroid y coordinates. Transition between lines are found as abrupt changes in y coordinates between adjacent regions. In this way y coordinates of text lines are determined and each ER is joined to the closest line. Line break is detected if difference between y coordinates of region centroids exceeds $threshold_{line\ break}$:

$$threshold_{line\ break} = \begin{cases} 0,4 * max(height[i], height[i-1]), \\ if\ heightRatio > 2 \\ 0,4 * mean(height[i], height[i-1]), \\ if\ heightRatio \leq 2 \end{cases} \quad (17)$$

$$heightRatio = \frac{max(height[i], height[i-1])}{min(height[i], height[i-1])} \quad (18)$$

This threshold allows characters to deviate from horizontal orientation what enables to handle certain degree of diagonal text orientation.
2. Regions in each text line are sorted by centroid x coordinate. Relation of each region to its neighboring components is checked using distance function proposed in [Yin14] which includes features like width and height difference, vertical and horizontal alignment etc. If distance value to both neighbors is higher than threshold (also defined in [Yin14]), ER is discarded as non-character object.

Character restoration step deals with problem of character regions erroneously discarded by SVM classifier. Candidates for restoration are ERs having low SVM probability score. For each region that already has been classified as character its adjacent restoration candidates are checked and restored if:

- Difference of centroid y coordinates between character region i and restoration candidate region j is lower than 50% of the character region height
- Horizontal distance between character region i and restoration candidate region j should satisfy:

$$\frac{H_{ij}}{mean(w_i, w_j)} < 2 \quad (19)$$

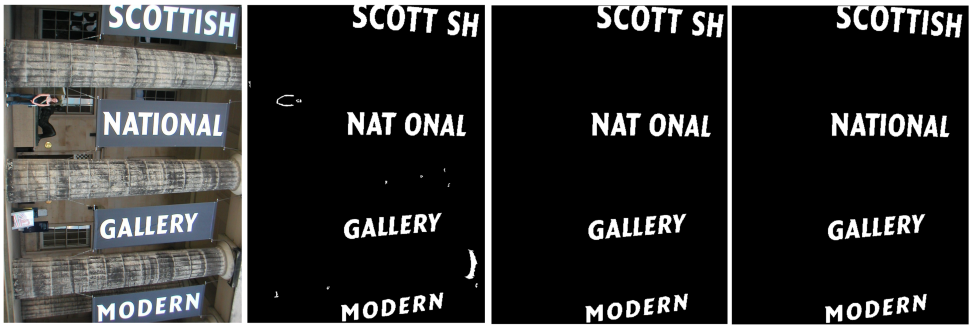


Figure 4: Character candidates grouping and restoration (from left to right): input image, image after SVM classification of character candidates, image after character candidates grouping and image after character candidates restoration

where H_{ij} is distance between right edge of left region and left edge of the right region, while w_i and w_j represent bounding box width for regions i and j.

- Restoration candidate is not contained inside character region
- Overlap of restoration candidate region with character region is lower than 20% of the character region area
- Value of distance function ([Yin14]) between restoration candidate region j and character region i is lower than threshold already used for character grouping

Example of character candidates grouping and restoration steps is shown on Figure 4.

3 EXPERIMENTAL RESULTS

Proposed method was evaluated on the dataset of ICDAR 2013 challenge "Reading text in scene images" [Kar13]. Same dataset is used in ICDAR 2015 challenge "Focused scene text" [Kar15]. Training set (229 images) and testing set (233 images) contain variety of scene text examples captured under challenging imaging conditions (low contrast, blur, uneven illumination, perspective distortion). Text segmentation performance is evaluated using precision and recall measures on pixel level and atom level. Since pixel-based measures don't reflect morphological properties of extracted region, atom-based metrics presented in [Cla10] is considered more relevant. It introduces minimal and maximal coverage criteria. The first criterion requires that extracted region should cover Tmin = 0.9 of a character skeleton. According to the second criterion, pixels outside the character area should be close to the character edge. More precisely, distance between pixel and edge should not exceed Tmax = max(5; 0.5*G), where G is the maximum stroke width of a character. Based on these criteria each region can be classified into following categories: Background,Whole, Fraction, Multiple,Fraction and Multiple,Mixed.

Table 1 compares performance of proposed text segmentation method with results presented in [Kar13] and [Lu15]. Proposed method achieves second highest precision and competitive recall. Higher precision value can be explained by influence of the proposed character generation method which uses redundant representation and multiple classifications of the same region. In this way elimination of non character candidates is more efficient. Proposed approach outperforms USTB_FuStar method which is based on algorithm described in [Yin14]. Proposed approach use simpler character candidates generation scheme without using maximum stability criterion for ER pruning. NSTsegmentator variant of method proposed in [Mil15] is significantly outperformed, while in comparison with NSTextractor variant of [Mil15] slightly lower F-score is achieved. Figure 5 shows examples of proposed scene text segmentation method. Method successfully extracted text in third example characterized by reflections and fourth image where complex background and windows structures were eliminated. In the first image error is caused by windows structures that are misinterpreted as character regions, while in the second image character "L" is lost due to highlights. Average running time per image calculated on ICDAR 2015 training set is about 14 seconds (Intel Core 2 Duo 3 GHz, 5 GB RAM). Execution time can be significantly improved by more efficient implementation in C++ and more recent CPU.

4 CONCLUSION

In this paper novel method for generation of character candidate regions is proposed. ER tree subpaths are constructed using area variation. Sufficiently long subpaths are taken as character candidates that are represented by subset of ERs from that subpath. Mean SVM probability score of ERs in this subset is used to decide whether subpath represent character or noise component. This step is followed by character grouping and restoration steps. Experimental results obtained on the ICDAR 2013 dataset show that the proposed

Table 1: Text segmentation results on ICDAR 2013 test set

	Pixel based results			Atom based results		
Method	Recall	Precision	F-Score	Recall	Precision	F-Score
Lu et al.[Lu15]	77.27 %	82.10 %	79.63 %	70.12 %	81.20 %	75.24 %
I2R_NUS_FAR	74.73 %	81.70 %	78.06 %	68.64 %	80.59 %	74.14 %
NSTextractor [Mil15]	60.71 %	76.28 %	67.61 %	63.38 %	83.57 %	72.09 %
Proposed method	60.32 %	79.93 %	68.76 %	64.43 %	81.21 %	71.85 %
USTB_FuStar [Yin14]	69.58 %	74.45 %	71.93 %	68.03 %	72.46 %	70.18 %
I2R_NUS	73.57 %	79.04 %	76.21 %	60.33 %	76.62 %	67.51 %
NSTsegmentator [Mil15]	68.41 %	63.95 %	66.10 %	68.00 %	54.35 %	60.41 %
Text Detection	64.74 %	76.20 %	70.01 %	62.03 %	57.43 %	59.64 %

Figure 5: Scene text segmentation examples

approach efficiently eliminates non character regions yielding high precision value and competitive recall value. Future work will focus on the improval of recall rate through integration of the sliding window approach after character candidates grouping, that is in the regions of interest corresponding to extracted text lines. In this way number of scanned windows will be reduced and scale would be known in advance. This approach would lower computational complexity of standard sliding window methods which requires processing of the whole image on multiple scales.

5 ACKNOWLEDGMENT

This work has been fully supported by Croatian Science Foundation under the project UIP-2014-09-3875.

6 REFERENCES

[Kar15] Karatzas D., Gomez-Bigorda L., Nicolaou A., Ghosh, S., Bagdanov, A, Iwamura, M.,Matas, J., Neumann, L. Icdar 2015 competition on robust reading, in Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, 1156-1160,2015.

[Jad14] Jaderberg, M:, Vedaldi, A., and Zisserman, A. Deep features for text spotting, in Computer Vision–ECCV 2014, 512-528, 2014.

[Wan12] Wang, T., David, J. W., Coates, A., and Ng, A.Y. End-to-end text recognition with convolutional neural networks, in Pattern Recognition (ICPR), 2012 21st International Conference on, 3304-3308, 2012.

[Yin14] Yin, X-C., Yin, X., Huang, K., Hao, H-W. Robust text detection in natural scene images. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 970-983, 36, No. 5, 2014.

[Neu15] Neumann, L., Matas, J. Real-time Lexicon-free Scene Text Localization and Recognition.

Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1872-1885, No. 99, 2015.

[Sun15] Sung, M-C., Jun, B., Cho, H., and Kim, D. Scene text detection with robust character candidate extraction method. Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, 426-430, 2015.

[Hua14] Huang, W., Qiao, Y., and Tang, X. Robust scene text detection with convolution neural network induced mser trees, in Computer Vision–ECCV 2014, 497-511, 2014.

[Lu15] Lu, S., Chen, T., Tian, S., Lim, J-H., and Tan, C-L. Scene text extraction based on edges and support vector regression. International Journal on Document Analysis and Recognition (IJDAR), 125-135, 18, No. 2, 2015.

[Mil15] Milyaev, S., and Barinova, O., Novikova, T., and Kohli, P., and Lempitsky, Victor. Fast and accurate scene text understanding with image binarization and off-the-shelf OCR, International Journal on Document Analysis and Recognition (IJDAR), 169-182, 18, No. 2, 2015.

[Mat04] Matas, J., and Chum, O., and Urban, M., and Pajdla, T. Robust wide-baseline stereo from maximally stable extremal regions, Image and vision computing, 761-767, 22, No. 10, 2004.

[Gon13] González, Á., and Bergasa, L. M. A text reading algorithm for natural images. Image and Vision Computing, 255-274, 31, No. 5, 2013.

[Sar16] SAric, M. Scene text segmentation using low variation extremal regions and sorting based character grouping, submitted to Neurocomputing, 2016

[Kar13] Karatzas D., et al. ICDAR 2013 robust reading competition, in Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, 1484-1493, 2013.

[Cla10] Clavelli, A., Karatzas, D., and Lladós, J. A framework for the assessment of text extraction algorithms on complex colour images, in Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, 19-26, 2010.

[Sha11] Shahab, A., Shafait, F., Dengel, A. ICDAR 2011 robust reading competition, in Document Analysis and Recognition (ICDAR), 2011 11th International Conference on, 1491-1496, 2011.

[Kai17] KAIST Scene Text Database. http://www.iapr-tc11.org/mediawiki/index.php?title=KAIST_Scene _ Text_ Database

[Cam09] De Campos, T.E., Babu, B.R., Varma, M. Character recognition in natural images, in VISAPP (2), pp. 273-280, 2009.

[Mar01] Martin, D., Fowlkes, C., Tal, D., and Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on, Vol. 2, IEEE, pp. 416-423, 2001.

Multi-references shape constraint for snakes

Mohamed Amine Mezghich
GRIFT Research Group,CRISTAL Laboratory, ENSI Campus Universitaire de la Manouba, Tunisia
amine.mezghich@ensi-uma.tn

Ines Sakly
GRIFT Research Group,CRISTAL Laboratory, ENSI Campus Universitaire de la Manouba, Tunisia
ines.sakly@ensi-uma.tn

Slim Mhiri
GRIFT Research Group,CRISTAL Laboratory, ENSI Campus Universitaire de la Manouba, Tunisia
slim.mhiri@ensi-uma.tn

Faouzi Ghorbel
GRIFT Research Group,CRISTAL Laboratory, ENSI Campus Universitaire de la Manouba, Tunisia
faouzi.ghorbel@ensi-uma.tn

ABSTRACT

In this research, we intend to present a new method of snakes with an invariant shape prior. We consider the general case where different templates are available and we have to choose the most suitable ones to define the shape constraint. A new external force is then proposed which is able to take into account several references at the same time with proportional weighting factors. Both a Fourier based shape alignment method and a complete and stable set of shape descriptors are used to ensure invariance and robustness of the prior knowledge to Euclidean transformations. To illustrate the efficiency of our approach, a set of experiments are applied on synthetic and real data. Promising results are obtained and commented.

Keywords

Snakes, shape constraint, complete and stable invariant descriptors, shape alignment.

1 INTRODUCTION

Image segmentation is a fundamental step in image processing. There have been tremendous researches for this problem. In many issues, segmentation implies a crucial role like in medical diagnosis, tracking and pattern recognition. Snakes have been widely used for object detection by a closed contour. The principle consists in moving a curve iteratively by minimizing an energy functional. The minimum is reached at object boundaries. Active contour methods can be classified into two families: parametric and level-set based active contours. The first family, called also snakes [Kass88, Cohen91, Xu98], uses an explicit representation of the contours to detect objects. The second one [Malladi95, Caselles97, Chan01] uses an implicit representation of the contours by level set approach to handle topological changes of the evolving front. Given that active contours in both families depend only on image gradient, there is still no way to characterize the global shape of an object. Especially in presence of occlusions and clutter, all the previous models converge to the wrong contours

For this purpose, many works introduced prior knowledge on the target object to be detected that can be geometric or statistic constraint on shape and/or texture.

Leventon & al. [Leventon00] associated a statistical shape model to the geodesic active contours [Caselles97]. Chen & al.[Chen01] defined an energy functional based on the quadratic distance between the evolving curve and the averge shapes of the target object after alignment. Bresson & al. [Bresson03] extended [Chen01] approach by integrating the statistical model of shape in the energy functional proposed by [Caselles97]. Fang & al. [Fang07] introduced a statistical shape prior into geodesic active contour to detect partially occluded object. PCA is computed on level set functions used as training data and the set of points in subspace is approximated by a Gaussian function to construct the shape prior model. To speed up the algorithm, an explicit alignment of shape prior model and the current evolving curve is done to calculate pose parameters. Mezghich & al. [Mezghich13] defined a new stopping function for the [Malladi95] model which is based on a reference shape after registration by phase correlation. For all the presented previous works, the authors use an edge-based level set active contour. Region-based approach is based on minimizing an energy functional to segment objects in the image. Experiments show that these models can detect objects with smooth boundaries and noise since the whole region is explored.

Foulonneau & al., [Foulonneau04] introduced a geometric shape prior into a region-based active contours [Chan01] based on the distance between the region inside the target and the reference shape. This distance is computed using the Legendre moments, which ensures invariance to translation and scaling. An extension of this work to the case of affine transformations is presented in [Foulonneau06]. Cremers &

al. [Cremers03] constructed a variational approach that incorporates a shape difference term into the same segmentation model. A labeling function is defined to indicate the regions in which shape priors should be enforced. Mezghich & al., [Mezghich12] defined also a new geometric shape prior based on curves alignment by Fourier transform. The model is able to detect partially occluded object under rigid transformations.
Like the level-set based models, many works introduced prior knowledge for snakes given that these models are well adapted for detection in real time (we have to manage only a reduced set of points). Staib & al., [Staib92] proposed to model shape by a Gaussian probability distribution. Optimization is performed using the maximum a posteriori based on Bayesian rule to define shape constraint. Diffusion Snakes [Cremers02] introduce a statistical prior knowledge on the evolving front to the Mumford-Shah model. Charmi & al. [Charmi08] use a complete and stable set of Fourier-based shape descriptors to define a new geometric constraint to the snake model which is invariant to Euclidean transformations and in [Charmi09], a shape alignment method is used as alternative to solve the problem of the starting point of the parametrized cure that represent the contour in the computation of descriptors.
In all the cited approaches, the template shape to be used is known in advance. In our knowledge, there have been a limited number of researches that treat the general case where we have a training data composed of several references and the model have to take into account the total information provided. In [Fang06] a statistical shape prior model is presented to give more robustness to object detection. This shape prior is able to manage different states of the same object, thus a Gaussian Mixture Model (GMM) and a Bayesian classifier framework are used. Using the level set functions for representing shape, this model suffer from the curse of dimensionality. Hence PCA was used to perform dimensionality reduction. In [Foulonneau09], a multi-references shape prior is presented for a region-based active contours. Prior knowledge is defined as a distance between shape descriptors based on the Legendre moments of the characteristic function of many available shapes.
Motivated by the two last presented works, we try in this research to propose a multi-references shape constraint for Snake. We construct a new statistical prior-driven term that is defined based on a given cluster of similar shapes according to the object to be detected. A complete and stable set of invariants descriptors is used to make the prior force parametrized with a studied weighting factors that give a significant sense. An alignment curves method is also used to guarantee invariance and robust point's curves matching. The improved model can retain all the advantages of snakes and have the additional ability of being able to handle the case of many references in presence of partial occlusions and noise.
The remainder of this paper is organized as follows : In Section 2, we will briefly recall the principle of the snake model. Then, we will devote Section 3 to present the used shape features for invariant shape description. In Section 4, the adopted shape alignment method based on Fourier transform is presented. Then, the multi-references shape constraint will be defined and explained in Section 5. A set of experimental results will be the focus of section 6. Finally, we conclude the work and highlight some possible perspectives in Section 7.

2 THE SNAKE MODEL

Active contours or Snakes [Kass88] consists of a parametrized curve $v(l) = (x(l), y(l))$ which moves under the influence of a functional energy until it reaches the edges of the object of interest. The functional energy is essentially composed of three terms:
- E_{int} : The internal energy which allows the smoothing of the contour and avoids the appearance of the corners,
- E_{image} : The external energy which pushes the curve towards the strong gradient of the image.
- E_{cont} : An external constraint forces that push the evolving curve to be similar to a given reference shape for example.
Hence, snake enery is as follows :

$$E^*_{snake} = \int_0^1 E_{int}(v(l)) + E_{image}(v(l)) + E_{cont}(v(l))dl, \tag{1}$$

Where E_{int} and E_{image} are defined as follows:

$$E_{int}(v(l)) = w_1|v^{'}(l)|^2 + w_2|v^{''}(l,t)|^2, \tag{2}$$

$$E_{image}(v(l)) = -w_3|\nabla(G_\sigma * I)|^2, \tag{3}$$

With w_1, w_2 and w_3 are a weighting factors of the different Snake's energy terms. I is the given image and G_σ is a Gaussian filter with a deviation equals to σ. The Snake equation was first solved by Kass & al. (See [Kass88], Appendix section, for a detailed description) using an Euler framework, which gives after discretization the following equation:

$$(I_N + t\,A)v(t) = v(t-1) + t\,F_{ext}(v(t-1)), \tag{4}$$

where A is a symmetric pentadiagonal matrix computed from the coefficients w_1 and w_2 of size N, representing all internal elasticity relations of the snake, t is the time step, N represents the number of points of the snake, I_N is the identity matrix and F_{ext} are the forces derived from the external energy:

$$F_{ext} = -\nabla|\nabla(G_\sigma * I)|^2, \tag{5}$$

3 SHAPE DESCRIPTION USING INVARIANT SET OF FOURIER DESCRIPTORS

Given that we will consider a cluster of closed curves which are similar to the evolving one, we adopt Fourier invariant description by a set of complete and stable features. The completeness property means that two objects have the same shape if and only if they have the same set of invariants. It allows also the reconstruction of shapes from their invariants up to an Euclidean transformation. Whereas the stability property means that a slight shape distortion does not induce a noticeable divergence of invariants. The used set of invariants Fourier descriptors (see [Ghorbel98] for more details) is as follows :

$$\begin{cases} I_{k_0}(\gamma) = |C_{k_0}(\gamma)|, \quad C_{k_0}(\gamma) \neq 0 \\ I_{k_1}(\gamma) = |C_{k_1}(\gamma)|, \quad C_{k_1}(\gamma) \neq 0, \quad k_1 \neq k_0 \\ I_k(\gamma) = \frac{C_k(\gamma)^{k_0-k_1} C_{k_0}(\gamma)^{k-k_1} C_{k_1}(\gamma)^{k_0-k}}{I_{k_0}^{k-k_1-p}(\gamma) I_{k_1}^{k_0-k-q}(\gamma)}, \\ k \neq k_0, k_1 \ \ and \ \ p,q > 0, \end{cases} \tag{6}$$

Where C_k is the well known Fourier descriptor. To compare shapes, we use the following distance between descriptors:

$$d(\gamma_1,\gamma_2) = \sum_k (|I_k(\gamma_1) - I_k(\gamma_2)|^{\frac{1}{2}})^2 \tag{7}$$

4 SHAPES ALIGNMENT USING FOURIER DESCRIPTORS

Curves alignment and matching are important steps for variability studies between shapes and the construction of prior knowledge. We use the method presented by Persoon & Fu [Persoon86] which is based on Fourier transform that makes it fast. We will recall the outline of this method. For a given planar object represented by it's parametric closed curve we have :

$$\begin{aligned} \gamma : [0, 2\pi] &\rightarrow C \\ t &\mapsto x(t) + i\, y(t), \end{aligned} \tag{8}$$

Let γ_1 and γ_2 be centred (according to the center of mass) and normalized arclength parameterizations of two closed planar curves having shapes F_1 and F_2. Suppose that γ_1 and γ_2 have a similar shape under rigid transformation. In shape space this is equivalent to have

$$\gamma_1(t) = \alpha e^{i\theta} \gamma_2(t + t_0), \tag{9}$$

where α is the scaling factor, θ the rotation angle, t_0 is the difference between the two starting points of γ_1 and γ_2. Rigid motion estimation is obtained by minimizing the following quantity:

$$E_{rr}(\gamma_1,\gamma_2) = min_{(\alpha,\theta,t_0)} ||\gamma_1(t) - \alpha e^{i\theta} \gamma_2(t + t_0)||_{L^2}, \tag{10}$$

Using Fourier descriptors $C_k(\gamma_1)$ and $C_k(\gamma_2)$, minimizing E_{rr} becomes equivalent to minimize $f(\theta,t_0)$ in Fourier domain

$$f(\theta,t_0) = \sum_{k \in Z} |C_k(\gamma_1) - \alpha e^{i(kt_0+\theta)} C_k(\gamma_2)|^2, \tag{11}$$

In [Persoon86], the authors proposed an analytical solution to compute t_0 and θ. t_0 is one of the zeros of the following function

$$\begin{aligned} g(t) = \sum_k \rho_k \sin(\psi_k + kt) \sum_k k\rho_k \cos(\psi_k + kt) - \\ \sum_k k\rho_k \sin(\psi_k + kt) \sum_k \rho_k \cos(\psi_k + kt), \end{aligned} \tag{12}$$

where $\rho_k\ e^{i\psi_k} = C_k^*(\gamma_1) C_k(\gamma_2)$, θ is chosen to satisfy Eq. (11) and minimize $f(\theta,t_0)$ where t_0 is one of the roots of g.

$$tan\theta = -\frac{\sum_k \rho_k \sin(\psi_k + kt_0)}{\sum_k \rho_k \cos(\psi_k + kt_0)}, \tag{13}$$

Having the value of θ and t_0 that minimize $f(\theta,t_0)$, the scaling factor is given by

$$\alpha = \frac{\sum_k \rho_k \cos(\psi_k + kt_0 + \theta)}{\sum_k C_k^*(\gamma_2) C_k(\gamma_2)}, \tag{14}$$

Based on Nyquist-Shannon theorem and B-spline approximation of the $g(t)$'s curve, the first implementation of this method and its application in rigid motion estimation for video compression was provided in [Ghorbel98]. In figure 1, we show an example of two parametrized curves of the same shape but under different rotation and scale factor. The right image shows

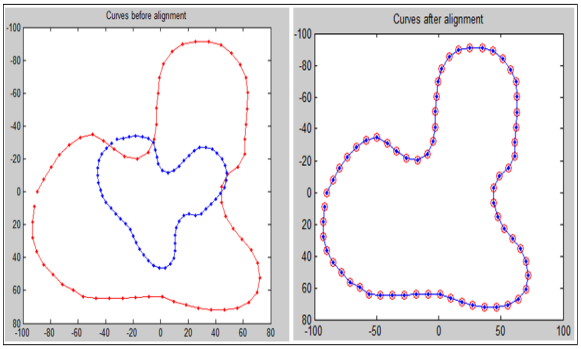


Figure 1: Two curves before and after alignment using the presented method.

results after alignment. Figure 2 represents the associated g function. The roots are draw with green circle.

This method will be used to align the training references to the active contour to define prior knowledge after the evolving curve being stable.

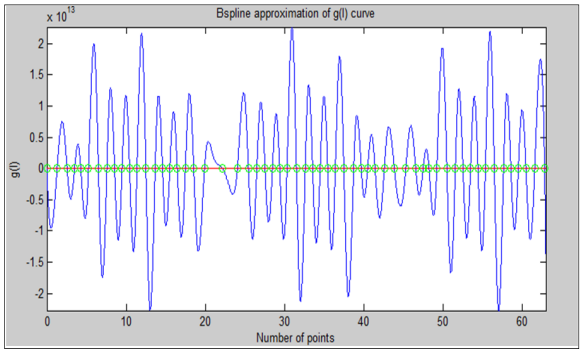


Figure 2: The correspondent g function

5 MUTLI-REFERENCES SHAPE CONSTRAINT FOR SNAKE

Consider a given set of N references having similar shape according to the evolving curve. It may be that among these templates, some of them are very similar to the target contour than the others. So convergence should be guided by this subset of curves. For this purpose, we need to defined a new weighting evolution schema that take into account this consideration. We presented in section 3 a distance $d(\gamma_1,\gamma_2)$, $(eq.7)$, between the invariant set of two curves. This distance will be used as criterion to study similarity between each reference and the target shape. It will provide us a mathematical way to sort the training data.
Let γ be the active contour and γ^i_{ref} be the i^{th} reference, The proposed prior force will be as follows:

$$F_{prior} = \sum_{i=1}^{N} g_i \cdot (\gamma - \gamma^i_{ref}), \qquad (15)$$

where $g_i = \frac{1}{1+d_i^p}, p > 1$ and $d_i = d(\gamma, \gamma^i_{ref})$.
As it can be seen, for a «near» reference γ^i_{ref}, we have $d_i^p \approx 0$, so $g_i \approx 1$, then this reference will be taken into account in the evolution process.
For a «far» reference having $d_i^p \approx \infty$, $g_i \approx 0$. So this reference will not influence the convergence to the desired shape. We can deduce that the proposed function g_i plays an important role to encourages evolution towards similar templates with natural weighting factor. We note that g_i recalls the edge-stopping function on high image gradient for the classic level set based active contours, see [Malladi95].
Hence the total evolution equation that we propose is :

$$(I_N + t\,A)v(t) = v(t-1) + t[c_1\,F_{ext}(v(t-1)) + c_2\,F_{prior}(v(t-1))], \qquad (16)$$

where c_1 and c_2 are two weighting factors related respectively to the data and the prior forces.

The proposed algorithm is then composed of two steps :

1- **Step 1 : Off-line step**
- It consists in the computation of invariants descriptors for each reference and its alignment according to a selected one.
2- **Step 2 : On-line step**
- Start the snake with only image driven-force.
- Having the contour being stable, we perform alignment of the N references according to the evolving front.
- We compute the proposed force F_{prior}
- Then, we evolve the model under both forces, the image and the prior ones.

6 EXPERIMENTAL RESULTS

To assess the robustness of the proposed approach, a set of experiments is performed. At first, we will evaluate the behavior of the new added force F_{prior} on simulated images. Then, we will experiment the proposed model on real images with partially occluded objects.

6.1 Results on simulated data

We consider in this first experiment a partially occluded object. Five references are considered to be added as prior knowledge (figure 3).

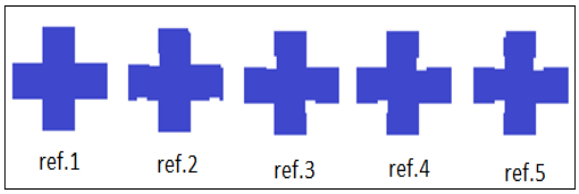


Figure 3: Some used references.

Each curve is sampled to 120 points. Results of segmentation are shown in figure 4. By the (a) image, we show the obtained detection using the classic snake model based on image driven forces.
In the (b) image, we present the segmentation using the snake model with shape prior. We use only one reference which is the closest one (here we get $ref.4$) according to the distance $d = d(\gamma, \gamma^4_{ref})$. This idea was proposed in [Charmi09].
By the (c) image, we show the obtained result with the proposed multi-references shape prior. Visually, it is clear that the detection using our model seems to be better than in (b). In fact, by our approach all the templates participate in the process of detection.
In (d), we present the aligned curves used as references.

By the table below, we present the obtained distance between each reference and the evolving curve. The weighing factors g_i are presented by the second row $(p = 2)$.

At the beginning, only snake forces are considered so we take $c_1 = 1$ and $c_2 = 0$. At a given iteration, when the evolving contour became stable, we introduce the

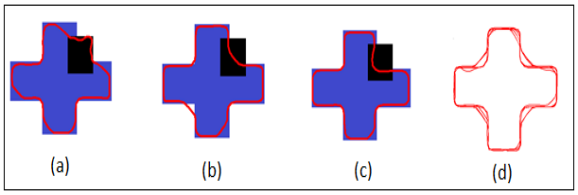


Figure 4: The obtained segmentation result with and without shape prior.

	Ref.1	Ref.2	Ref.3	Ref.4	Ref.5
$d(\gamma, \gamma^i_{ref})$	2.183	1.938	2.382	1.899	2.232
g_i	0.209	0.266	0.176	0.277	0.200

Table 1: The weighing factors used for each reference.

prior force F_{prior} using $c_1 = 0.7$ and $c_2 = 0.3$.
We end this part by testing our proposed model on partially occluded object under noise, missing part and clutterd background. We use the same configuration as for the previous experiment. Results are show by figure 5.

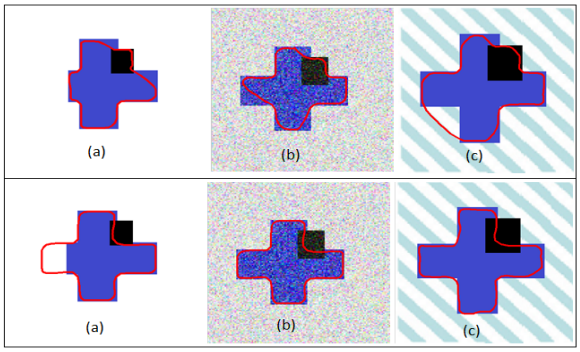


Figure 5: Object detection under missing part (a), noise (b) and clutter (c).

In what follows, the proposed model is applied to real data. Two applications are considered: the first one consists in the tracking of moving object under cluttered background and by the second experiment, we try to detect the left ventricle of the heart in myocardial scintigraphy slices under low contrast.

6.2 Application 1 : Tracking of moving object under clutter background

The object to be detected presents deep concavity which is unresolved problem by the classic snake model. Besides, the background is cluttered. As it ca be seen, our model succeeds in finding the true contours of the target object.
We used for each row 4 references to construct prior shape knowledge. For the first row, the contour is sampled to 200 points, and for the second one, we used 400 points. Finally, for the last one, we take 300 points. Figure 7 shows the used references for each object after sampling and alignment according to the evolving curve.

Figure 6: First column : Initial contour, Second column : Result without prior knowledge, Last column : Result with prior knowledge.

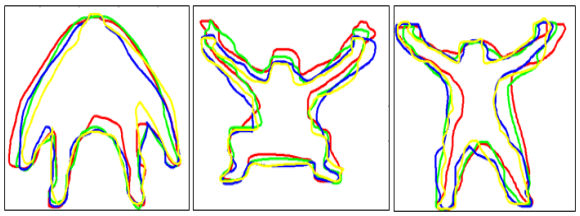


Figure 7: The used references.

6.3 Application 2 : Segmentation of myocardial scintigraphy images

For this type of image, usually object of interest are in low contrast, so snakes fail to detect the target object as it is show in image (b) of figure 8. Using references (provided by the expert) associated to the previous and the next slices according to the current one, result obtained by the proposed model are presented by image (c). As it is shown, results seems to be encourages using only 2 references.

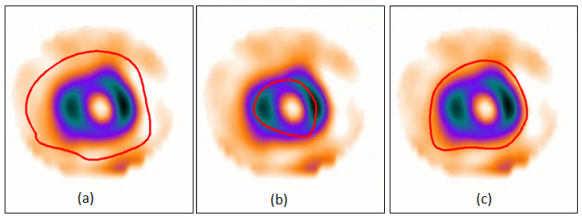


Figure 8: (a):initial contour; (b):segmentation without prior; (c):segmentation with prior knowledge.

7 CONCLUSION

A new shape constraint for the snake model is presented in this paper. The method is able to manage several references with a studied weighting factors under invariance to rigid transformations. Results show robust object detection under partial occlusion, clutter and noise.

Besides, given that the method is based on Fast Fourier Transform (FFT) for shape alignment and descriptors computation, the method can be applied for tracking moving object in real time.The medical context represents also a challenging field in future work.

8 REFERENCES

[Kass88] M. Kass, A. Witkin, and D. Terzopoulos, Snakes: active contour models, Int. J. of Comp. Vis., vol.1, no.4, pp. 321-331, 1988.

[Cohen91] Cohen, L., On active contour models and balloons, In Graphical Models Image Process, 1991.

[Xu98] C. Xu and J. Prince, Snakes, shapes, and gradient vector flow, IEEE trans. IP, vol. 7(3), pp. 359-369, 1998.

[Malladi95] Malladi, R., Sethian, J., and Vemuri, B., Shape modeling with front propagation: A level set approach, In IEEE Trans. Patt. Anal. Mach. Intell., 1995.

[Caselles97] Caselles, V., Kimmel, R., and Sapiro, G., Geodesic active contours, In Int. J. of Comp. Vis., 1997.

[Chan01] Chan, T. and Vese, L., ctive contours without edges, A. In IEEE Trans. Imag. Proc, 2001.

[Charmi08] M. A. Charmi, S. Derrode, and F. Ghorbel, Fourier based shape prior for snakes, Pat. Recog. Let., vol.29(7), pp. 897-904, 2008

[Charmi09] M. A. Charmi, S. Derrode, and F. Ghorbel, Using Fourier-based shape alignment to add geometric prior to snakes, ICASSP, 2009.

[Cremers02] D. Cremers, F. Tischhauser, J. Weickert, and C. Schnorr, Diffusion snakes: introducing statistical shape knowledge into the Mumford-Shah functional, Int. J. of Comp. Vis., vol. 50, pp. 295-313, 2002.

[Staib92] L.H. Staib and J.S. Duncan, Boundary finding with parametrically deformable models, IEEE trans. PAMI,vol. 14, no. 11, pp. 1061-1075, 1992.

[Persoon86] E. Persoon and K. S. Fu, Shape discrimination using Fourier descriptors, IEEE trans. PAMI, vol. 8, no. 3, pp. 388-397, 1986.

[Ghorbel98] F. Ghorbel, Towards a unitary formulation for invariant image description: application to image coding, An. of telecom., vol. 153, no. 3, pp. 145-155, 1998.

[Fang06] Fang, W. and Chan, K., Using statistical shape priors in geodesic active contours for robust object detection, In ICPR, 2006.

[Fang07] Fang, W. and Chan, K., Incorporating shape prior into geodesic active contours for detecting partially occluded object, In Pattern Recognition, 2007.

[Leventon00] M. Leventon, E. Grimson and O. Faugeras, Statistical shape influence in geodesic active contours. Proc. of IEEE Conference on Computer Vision and Pattern Recognition, pp.316-323, 2000.

[Chen01] Chen, Y., Thiruvenkadam, S., Tagare, H., Huang, F., Wilson, D., and Geiser, E., On the incorporation of shape priors into geometric active contours. In IEEE Workshop on Variational and Level Set Methodsin Computer Vision., 2001.

[Cremers03] Cremers, D., Sochen, N., and Schnorr, C., Towards recognition-based variational segmentation using shape priors and dynamic labelling, In International Conference on Scale Space Theories in Computer Vision, 2003.

[Bresson03] Bresson, X., Vandergheynst, P., and Thiran, J., A priori information in image segmentation : energy functional based on shape statistical model and image information. In Proc. of IEEE Conference on Image Processing., 2003.

[Foulonneau04] Foulonneau, A., Charbonnier, P., and Heitz, F., Contraintes geomtriques de formes pour les contours actifs orientes region : une approche basee sur les moments de legendre. In Traitement du signal, 2004.

[Foulonneau06] Foulonneau, A., Charbonnier, P., and Heitz, F., Affine-invariant geometric shape priors for regionbased active contours. In IEEE Trans. Patt. Anal. Mach. Intell., 2006.

[Foulonneau09] Foulonneau, A., Charbonnier, P., and Heitz, F., Multi-Reference Shape Priors for Active Contours. In International Journal of CComputer Vision. vol. 81:68, 2009.

[Mezghich12] M-A. Mezghich, S. M'hiri and F. Ghorbel, Fourier-based multi-references shape prior for region-based active contours. 16th IEEE Mediterranean Electrotechnical Conference (MELECON), p.661-664 2012.

[Mezghich13] M-A. Mezghich, M. Sellami, S. M'hiri and F. Ghorbel, Shape prior for an edge-based active contours using phase correlation. EUSIPCO,2013.

[Mezghich14] M-A. Mezghich, S. M'hiri and F. Ghorbel, Invariant Shape Prior Knowledge for an Edge-based Active Contours. VISAPP, p. 454-461, 2014.

[Mezghich17] M-A. Mezghich, S. M'hiri and F. Ghorbel, Robust statistical prior knowledge for active contours. VISAPP, Accepted. 2017.

Semantic Search User Interface Patterns : An Introduction

Edgard Marx
University of Leipzig
Augustusplatz 10
04109, Leipzig, Saxony
marx@informatik.uni-leipzig.de

Ali Khalili
VU University Amsterdam
The Netherlands
a.khalili@vu.nl

André Valdestilhas
University of Leipzig
Augustusplatz 10
04109, Leipzig, Saxony
valdestilhas@studserv.uni-leipzig.de

ABSTRACT

Within the past few years, many patterns and principles have been proposed towards the enhancement of search user interfaces and experience. However, to access and explore information efficiently is still significantly challenging. Recently, we have seen the rise of a new kind of information retrieval approach, the so-called semantic search systems. These systems promise more accurate results while exploring semantics of the data. Although there exist several search user interfaces tailored to semantic search, there is still a lack of usability studies as well as good practices. In this work, we discuss the applicability of traditional search user interfaces in semantic search systems. Furthermore, we propose a new interaction model based on four patterns: Poli-Communicative, Discrete Display, Heterogeneous Data-face and Dive in-place.

Keywords

Semantic Search User Interface, Interaction Design, Human-Computer Interaction, Semantic Search

1 INTRODUCTION

Although significant efforts have been devoted to research and development of search engines, the search is not a solved problem. In fact, users still find it challenging to access and explore information. With the advent of the Internet, personal computers, mobile devices as well as Smart TVs, the amount of information generated by users is increasing day by day. More and more users resort to search engines to find the information needed. Indeed, search engines are shaping how we access and learn information. However, apart from the sophisticated algorithms behind search engines, there are other aspects as important as a good search algorithm: the user interface.

User interfaces are the gateway of the search engines, they facilitate users to find, explore, and understand the information. Since the inception of the first search engine [3], many ideas have been developed to enable and facilitate content access and exploration. Some of these ideas have become repeatable good practices to solve common problems and thereby have been established as patterns. Examples of such patterns are Faceted Search and Autocomplete.

In this work, we discuss the applicability of search patterns and propose an extension of those for semantic search interfaces. By *semantic search*, we do not restrict our view to engines that make use of RDF data, but to applications that try to understand the searcher's intent by using the contextual meaning of the terms in the query. Moreover, we extend the concept of semantics to visual aspects of the results been displayed to the user. As it is shown by many researchers [5, 8, 9], these aspects do influence the user's cognitive perception and thus should be explored. Our aim is to develop a semantic search user interface as part of the openQA framework [7]. openQA is a framework designed for fast and easy development of questions answering and semantic search approaches. The remaining of this paper is structured as follows. Section 2 discusses four patterns for semantic search that we propose. Finally, Section 3 concludes with an outlook of the future work.

2 A SEMANTIC SEARCH USER INTERFACE

In this section, we present and illustrate four patterns – (1) Poli-Communicative, (2) Discrete Display, (3) Heterogeneous Data-face and (4) Dive in-place – that extend the ten previously introduced patterns. Some of the proposed patterns are (partially) implemented in some of the existing semantic search applications, but they were not previously defined [8]. The four patterns are based on user experience as well as an analytical review of the literature.

We propose that semantic search interfaces should not limit themselves to merely support Natural Language techniques such as query expansion or better understanding complex queries, but be poli-communicative. That is, displaying the information more efficiently, embracing users with cognitive difficulties as well as the many aspects of the communication process.

2.2 Discrete Display

Information can be more or less important for a given input query. Krug et al. [5] argue that users spend a little time reading most Web pages. Instead, *they scan them, looking for words or phrases that catch their eye*. Therefore, Krug concludes that clear visual hierarchy is *the best way to make a page easy to grasp in a hurry*. Most of the semantic search interfaces restrict the differentiation of result relevance by its position in the resulting list. However, the layout is very important for user's interaction and perception, it has to be with the organization of the data being displayed. We propose that the relevance of an information should not just affect its position, but also its style as well as the mode it is being displayed with. For instance, data with more importance should be displayed with more details and more evidence (big fonts, big boxes, big images) than other data. *Google*'s interface provides these capabilities to some extent. For instance, when displaying the result for the query "Michael Schumacher" Google shows the required information on the right while the documents sorted by relevance are on the left side. However, the user might be looking for the result in the right, but when displaying documents on the left, Google is clearly giving more emphasis to the documents.

This design pattern clears is incosistent with usability studies [2, 9, 4][2] that shows that user's attention follows the same read/write patterns (F-Shaped Pattern). That is, the user's attention goes descending from left to right and top to bottom. Furthermore, the documents are being displayed with the same font and emphasis. The only apparent difference is the position in the result list.

Figure 2 shows one of most emblematic examples of Discrete Display, the newspaper. The newspaper displays information accordingly with its relevance. The most relevant information occupied more space, have bigger fonts and appears on the first pages. Figure 1(a) depicts an example of (1) Discrete Display for semantic search interfaces in which the most relevant result is positioned on top with more evidence.

[2] https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content

Figure 2: Example of Discrete Display pattern in the daily journal New York Times.

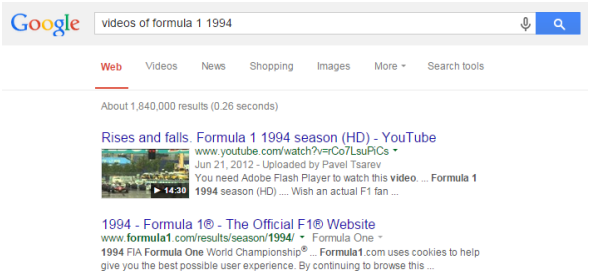


Figure 3: Layout 3. Google displaying results for 1994 F1 videos.

2.3 Heterogeneous Data-face

A good semantic search interface should support different data presentation. Most of the data semantic search interfaces do not take into consideration the type of data being displayed. Heterogeneous Data-face (HD) is an extension of the pattern Structured Results. The main difference is that Structured Results is about having structured data in the result page mainly focusing on user's intent, while HD is about displaying results from different data sources in the resulting page–e.g. video, audio, documents, structured information, and image. Data can have different types and properties, thereby are heterogeneous. In this sense, a good semantic search interface must support different types of data presentation. Many search interfaces already explore this concept such as Google, Yahoo, and Bing. For instance, geographic data can be displayed on a map while video can be displayed in a video canvas. However, there is still a place for improvement. One example is the result displayed in Google for the query "videos of formula one"—Figure 3—in which there are documents instead of the required videos.

The problem is that—for the previous query "videos of formula one"—there is a need to switch between different tabs (Web, Video, among others) in order to obtain the required information. One of the big challenges is how to improve the way the content is being displayed. Although there are different contents related to the query (e.g. Video, Web pages, and Structured Data), the top one structured information is displayed on the right and the top ten documents are listed on the left. These patterns often repeat between semantic

search engines, but: (1) Why not display videos, images and other content related to the query? and; (2) Why—instead of displaying the top one structured data and top ten web pages—search engines do not display the top ten results, independently of the data type?

We propose that semantic search interfaces should have a heterogeneous data-face. Figure 1(a) shows a semantic search interface displaying heterogeneous results, an example of the pattern (2) HD. The given example also implements a better version of Actionable Results than the one used by Google in Figure 3. Different from the previous models, it can promote a better interaction experience to users because all query related contents are displayed in a single canvas where it can also be activated.

2.4 Dive in-place

Dive in-place is an extension of the pattern Actionable Results. The idea behind this concept is that the interface should be self-contained. That is, the user should be allowed to explore the displayed information in-place. Dive in-place is not the same as Faceted Search. Faceted Search necessarily involves a change of the elements in the view and is also known as Faceted Navigation or Faceted Browsing, which involves the application of filters. Dive in-place is followed close by the design principle of Cooper et al. [1] that emphasizes as a good design principle the reduction of *places to go*. According to Cooper, reducing the number of screens, pages, and modes on Web sites increases people's ability to stay oriented.

Dive in-place involves the addition of new elements, it allows a more detailed version of the content without leaving the view. Some applications implement this pattern as a modal window or as a *more* button, where a detailed content is revealed to the user. However, a good example of Dive in-place is the Google's *Image* tab which allows users to interact with the images by giving an expanded version in-place. Different from the *Image* tab, a bad example is the *Map* tab. When a user requests the *Map* tab of the current search, the content is displayed in a new page. A good behavior is to open an extended view in-place in the same canvas in which the user can interact with the content. Search engines, usually force users to open the content in different windows in order to find the information needed.

By switching back and forth in order to find the information, the exploration can become tedious. Dive in-place is a good pattern to avoid such an experience. Figure 1(b) demonstrates the pattern Dive in-place in action. In the given interface, an extended view of the page content is shown enabling further exploration.

3 CONCLUSION AND FUTURE WORK

In this work, we presented four patterns for semantic search interface: (i) Poli-Communicative, (ii) Discrete Display, (iii) Heterogeneous Data-face and (iv) Dive in-place. As future work, we plan to do an extensible and detailed evaluation as well literature review. We see this work as the first step towards the enhancement of semantic search user interfaces.

4 ACKNOWLEDGMENTS

This work was partly supported by a grant from the German Research Foundation (DFG) for the project *Professorial Career Patterns of the Early Modern History: Development of a scientific method for research on online available and distributed research databases of academic history* under the grant agreement No GL 225/9-1, by CNPq under the program *Ciências Sem Fronteiras* process 200527/2012-6.

5 REFERENCES

[1] Alan Cooper and Robert Reimann. *About Face 3.0: The Essentials of Interaction Design*. Wiley & Sons, 3rd revised edition edition, 2007.

[2] Edward Cutrell and Zhiwei Guan. What are you looking for?: an eye-tracking study of information usage in web search. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 407–416. ACM, 2007.

[3] Alan Emtage and Peter Deutsch. Archie: An electronic directory service for the internet. In *Proceedings of the Winter 1992 Usenix Conference*, pages 93–110, 1992.

[4] Laura Granka, Matthew Feusner, and Lori Lorigo. Eyetracking in online search.

[5] Steve Krug. *Don't Make Me Think : A Common Sense Approach to the Web*. New Riders Publishing, Berkeley, Calif., 2. ed. edition, 2005.

[6] Yang Li. Gesture Search: A Tool for Fast Mobile Data Access. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 87–96, New York, NY, USA, 2010. ACM.

[7] Edgard Marx, Ricardo Usbeck, Axel-Cyrille Ngomo Ngonga, Konrad Höffner, Jens Lehmann, and Sören Auer. Towards an Open Question Answering Architecture. In *SEMANTiCS 2014*, 2014.

[8] Peter Morville and Jefferey Callender. *Search Patterns - Design for Discovery.* O'Reilly, 2010.

[9] Jakob Nielsen and Kara Pernice. *Eyetracking web usability*. New Riders, 2010.