The 13-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2005

in co-operation with

EUROGRAPHICS

W S C G ' 2005

Posters

University of West Bohemia
Plzen
Czech Republic

Honourary Chair

M.L.V.Pitteway, Brunel University, Uxbridge, United Kingdom

Co-Chairs

Tosiyasu L. Kunii: Kanazawa Institute of Technology, Tokyo, Japan
Vaclav Skala, Univ. of West Bohemia, Plzen, Czech Republic

Edited by

Vaclav Skala

WSCG'2005 Posters Proceedings

Editor-in-Chief: Vaclav Skala
University of West Bohemia, Univerzitni 8, Box 314
306 14 Plzen
Czech Republic
skala@kiv.zcu.cz
Managing Editor: Vaclav Skala

Author Service Department & Distribution:
Vaclav Skala - UNION Agency
Na Mazinách 9
322 00 Plzen
Czech Republic

Printed at the University of West Bohemia

Hardcopy: *ISBN 80-903100-8-7*

WSCG 2005

International Programme Committee

Alexa, Marc (Germany)
Bajaj, Chandrajit (United States)
Bartz, Dirk (Germany)
Bekaert, Philippe (Belgium)
Benes, Bedrich (Mexico)
Bengtsson, Ewert (Sweden)
Bouatouch, Kadi (France)
Brodlie, Ken (United Kingdom)
Brunet, Pere (Spain)
Brunnet, Guido (Germany)
Clapworthy, Gordon (United Kingdom)
Coquillart, Sabine (France)
Debelov, Victor (Russia)
Deussen, Oliver (Germany)
du Buf, Hans (Portugal)
Ertl, Thomas (Germany)
Ferguson, Stuart (United Kingdom)
Floriani, Leila De (Italy)
Flusser, Jan (Czech Republic)
Goebel, Martin (Germany)
Haber, Jörg (Germany)
Harris, Mark (United Kingdom)
Hauser, Helwig (Austria)
Hege, Hans-Christian (Germany)
Chen, Min (United Kingdom)
Chrysanthou, Yiorgos (Cyprus)
Jansen, Frederik,W. (The Netherlands)
Jorge, Joaquim (Portugal)
Kakadiaris, Ioannis (United States)
Kalra, Prem (India)
Kjelldahl, Lars (Sweden)
Klein, Reinhard (Germany)
Klosowski, James T. (United States)
Kobbelt, Leif (Germany)
Kruijff, Ernst (Germany)
Magnor, Marcus (Germany)
Margala, Martin (United States)
Moccozet, Laurent (Switzerland)
Mudur, Sudhir,P. (Canada)
Mueller, Klaus (United States)
Muller, Heinrich (Germany)
Myszkowski, Karol (Germany)
O'Sullivan, Carol (Ireland)
Pasko, Alexander (Japan)
Peroche, Bernard (France)
Post, Frits H. (Netherlands)
Puech, Claude (France)
Puppo, Enrico (Italy)
Purgathofer, Werner (Austria)
Rauterberg, Matthias (Netherlands)
Rheingans, Penny (United States)
Rokita, Przemyslaw (Poland)
Rossignac, Jarek (United States)
Rudomin, Isaac (Mexico)
Sbert, Mateu (Spain)
Shamir, Ariel (Israel)
Schaller, Nan,C. (United States)
Schneider, Bengt-Olaf (United States)
Schumann, Heidrun (Germany)
Skala, Vaclav (Czech Republic)
Slusallek, Philipp (Germany)
Sochor, Jiri (Czech Republic)
Stuerzlinger, Wolfgang (Canada)
Sumanta, Pattanaik (United States)
Szirmay-Kalos, Laszlo (Hungary)
Taubin, Gabriel (United States)
Teschner, Matthias (Switzerland)
Theoharis, Theoharis (Greece)
Trahanias, Panos (Greece)
Velho, Luiz (Brazil)
Veltkamp, Remco (Netherlands)
Weiskopf, Daniel (Germany)
Westermann, Ruediger (Germany)
Wuethrich, Charles Albert (Germany)
Zara, Jiri (Czech Republic)
Zemcik, Pavel (Czech Republic)

WSCG 2005 Board of Reviewers

Adzhiev,V. (United Kingdom)
Alexa,M. (Germany)
Ammann,C. (Switzerland)
Anan,H. (United States)
Andreadis,I. (Greece)
Artusi,A. (Italy)
Aspragathos,N. (Greece)
Aveneau,L. (France)
Bajaj,C. (United States)
Bartz,D. (Germany)
Bekaert,P. (Belgium)
Benes,B. (Mexico)
Bengtsson,E. (Sweden)
Bieri,H. (Switzerland)
Bilbao,J. (Spain)
Bischoff,S. (Germany)
Bottino,A. (Italy)
Bouatouch,K. (France)
Bourdin,J. (France)
Brodlie,K. (United Kingdom)
Brunet,P. (Spain)
Brunnet,G. (Germany)
Buehler,K. (Austria)
Callieri,M. (Italy)
Clapworthy,G. (United Kingdom)
Coleman,S. (United Kingdom)
Coombe,G. (USA)
Coquillart,S. (France)
Daniel,M. (France)
de Aquiar,E. (Germany)
De Decker,B. (Belgium)
de Geus,K. (Brazil)
Debelov,V. (Russia)
del Rio,A. (Germany)
Deussen,O. (Germany)
Diehl,S. (Germany)
Dingliana,J. (Ireland)
Dmitriev,K. (Germany)
Doleisch,H. (Austria)
Dong,F. (United Kingdom)
Drakopoulos,V. (Greece)
du Buf,H. (Portugal)
Duce,D. (United Kingdom)
Durupina,F. (Turkey)
Egges,A. (Switzerland)
Eibl,M. (Germany)
Erbacher,R. (United States)
Ertl,T. (Germany)
FariaLopes,P. (Portugal)
Faudot,D. (France)
Feito,F. (Spain)
Ferguson,S. (United Kingdom)
Fernandes,A. (Portugal)
Fischer,J. (Germany)
Flaquer,J. (Spain)
Floriani,L. (Italy)
Flusser,J. (Czech Republic)
Gagalowicz,A. (France)
Galo,M. (Brazil)
Geraud,T. (France)
Giannini,F. (Italy)
Gudukbay,U. (Turkey)
Gutierrez,D. (Spain)
Haber,J. (Germany)
Hadwiger,M. (Austria)
Haro,A. (United States)
Harris,M. (United Kingdom)
Hast,A. (Sweden)
Hauser,H. (Austria)
Havran,V. (Germany)
Hege,H. (Germany)
Hladuvka,J. (Slovakia)
Horain,P. (France)
Hornung,A. (Germany)
Chen,M. (United Kingdom)
Chin,S. (Korea)
Chover,M. (Spain)
Chrysanthou,Y. (Cyprus)
Iwanowski,M. (Poland)
Jaillet,F. (France)
Jansen,F. (Netherlands)
Jeschke,S. (Germany)
JoanArinyo,R. (Spain)
Kalra,P. (India)
Kjelldahl,K. (Sweden)
Klosowski,J. (United States)
Kobbelt,L. (Germany)
Kolcun,A. (Czech Republic)
Koutek,M. (Netherlands)
Krolupper,F. (Czech Republic)
Kruijff,E. (Germany)
Larsen,B. (Denmark)

Leopoldseder,S. (Austria)
Lewis,J. (United States)
Lintu,A. (Germany)
Loizides,A. (Cyprus)
Loizides,A. (Cyprus)
Magnor,M. (Germany)
Maierhofer,S. (Austria)
Mandl,T. (Germany)
Mantler,S. (Austria)
Margala,M. (United States)
Marinov,M. (Germany)
Maughan,C. (USA)
McAllister,D. (USA)
McMenemy,K. (United Kingdom)
Mertens,T. (Belgium)
Moccozet,L. (Switzerland)
Mokhtari,M. (Canada)
Moltedo,L. (Italy)
Montrucchio,B. (Italy)
Moreton,H. (USA)
Mudur,S. (Canada)
Mueller,K. (United States)
Muller,H. (Germany)
Myszkowski,K. (Germany)
Neubauer,A. (Austria)
Nielsen,F. (Japan)
O'Sullivan,C. (Ireland)
Ozguc,B. (Turkey)
Pan,Z. (China)
Pandzic,I. (Croatia)
Pasko,A. (Japan)
Pedrini,H. (Brazil)
Perez,M. (Spain)
Peroche,B. (France)
Plemenos,D. (France)
Post,F. (Netherlands)
Prakash,E. (Singapore)
Pratikakis,I. (Greece)
Prikryl,J. (Czech Republic)
Puppo,E. (Italy)
Purgathofer,W. (Austria)
Rauterberg,M. (Netherlands)
Renaud,c. (France)
Revelles,J. (Spain)
Rheingans,P. (United States)
Rodrigues,M. (United Kingdom)
Rokita,P. (Poland)
Rossignac,J. (United States)
Rudomin,I. (Mexico)
Sainz,M. (USA)
Sbert,M. (Spain)
Segura,R. (Spain)
Shamir,A. (Israel)
Schaller,N. (United States)
Schneider,B. (United States)
Scholz,V. (Germany)
Schumann,H. (Germany)
Sijbers,J. (Belgium)
Sips,M. (Germany)
Sirakov,N. (United States)
Sitte,R. (Australia)
Slusallek,P. (Germany)
Snoeyink,J. (United States)
Sochor,J. (Czech Republic)
Sorel,M. (Czech Republic)
Sroubek,F. (Czech Republic)
Stuerzlinger,W. (Canada)
Stylianou,G. (Cyprus)
Suarez Rivero,J. (Spain)
Sumanta,P. (United States)
Szekely,G. (Switzerland)
Szirmay-Kalos,L. (Hungary)
Tang,W. (United Kingdom)
Taubin,G. (United States)
Teschner,M. (Germany)
Theobald,C. (Germany)
Theoharis,T. (Greece)
Theußl,T. (Austria)
Tobler,R. (Austria)
Torres,J. (Spain)
Trahanias,P. (Greece)
Traxler,A. (Austria)
Van Laerhoven,T. (Belgium)
Velho,L. (Brazil)
Veltkamp,R. (Netherlands)
Vergeest,J. (Netherlands)
Vuorimaa,P. (Finland)
Weiskopf,D. (Germany)
Weiss,G. (Germany)
Westermann,R. (Germany)
Wu,S. (Brazil)
Wuethrich,C. (Germany)
Yilmaz,T. (Turkey)
Zach,C. (Austria)
Zachmann,G. (Germany)
Zara,J. (Czech Republic)
Zemcik,P. (Czech Republic)
Zhu,Y. (United States)
Zitova,B. (Czech Republic)

WSCG 2005

Poster proceedings

Contents

Session - W

SVG Rendering of Digital Images: an Overview

Sebastiano Battiato, Gianpiero Di Blasi,
Giovanni Gallo, Salvo Nicotra
Dipartimento di Matematica ed Informatica
Università di Catania - Viale A. Doria 6 – 95125 Catania Italy
{battiato, gdiblasi, gallo, snicotra} @dmi.unict.it

Giuseppe Messina
STMicroelectronics – AST Catania Lab Imaging Group - FAB. M6, Contrada Blocco Torrazze, Casella Postale 421, 95121 Catania, Italy
giuseppe.messina@st.com

ABSTRACT
The SVG (Scalable Vector Graphics) standard allows representing complex graphical scenes by a collection of graphic vectorial-based primitives, offering several advantages with respect to classical raster images such as: scalability, resolution independence, etc. In this paper we present a full comparison between some advanced raster to SVG algorithms: SWaterG, SVGenie, SVGWave and Vector Eye. SWaterG works by a watershed decomposition coupled with some ad-hoc heuristics, SVGenie and SVGWave use a polygonalization based respectively on Data Dependent and Wavelet triangulation, while Vector Eye is a commercial tool. Final quality has been assessed over a large dataset of images both in terms of PSNR and compression ratio.

Keywords
SVG, Triangulation, Watershed, Wavelet, Vectorialization.

1. INTRODUCTION
SVG is a language for describing two-dimensional graphics and graphical applications in XML ([6], [10]). In this work we are interested in reviewing existing techniques devoted to cover the gap between the graphical vectorial world and the raster world typical of digital photography. SVG format could find useful application in the world of mobile imaging devices, where typical camera capabilities should match with limited color/size resolutions displays. The core of the current SVG developments is the version 1.1. Actually the SVG 1.2 specification is under development and available in draft form. Major details can be found directly at the W3C site [12]. An exhaustive overview of the recent SVG development and related applications can be found in the proceedings of SVG Open Conference [12].
Two advanced techniques SVGenie [3] and SVGWave [5] have been applied to approximate local pixel neighborhood by triangles: the Data Dependent Triangulation (DDT) ([7]), the Wavelet Based Triangulation (WBT) ([9]).
The DDT replaces the input image with a set of triangles according to a specific cost function able to implicitly detect the edge details. The overall perceptual error is then minimized choosing a suitable cost function able to simplify triangulation. Recently further optimization of such function has been introduced for Colour Filtering Array demosaicing ([11]) and for image interpolation. The WBT uses the wavelet multilevel transformation to properly extract the details from the input images; a reverse process of triangulation, starting from the lowest level, is applied to derive the final WBT. A triangulation is, by first, achieved at the lowest level, introducing large triangles; then the process is refined by iterating for each level, the level details of each single triangle, according to the wavelet transformation. Both these decomposition could be directly managed by SVG primitives. Although the quality achieved in this way is rather good the size of the resulting files may be very large. The triangulation of both DDT and WBT are then processed by the polygonalization. The main purpose of this step is to minimize the dimensions of the resulting files by merging triangles together, according to specific similarity metrics, reducing the amount of perceptual redundancies. Another recent approach uses a raster-to-vector technique SWaterG ([4]) by advanced watershed decomposition ([15]) coupled with some ad-hoc heuristics devoted to obtain high quality rendering.

POSTERS proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

(a) (b) (c) (d)

Figure 1**: A comparison between some advanced SVG rendering (scale 4:1):** (a) **SVGenie,** (b) **SVGWave,** (c) **Vector Eye,** (d) **SWaterG.**

The system is composed by two main steps: first, the image is partitioned into homogeneous and contiguous regions using watershed decomposition. Secondly, an SVG representation of such areas is achieved by ad-hoc chain code building.

In [1] a conversion tool able to obtain SVG representation of Geomatics data has been presented but the overall results are limited by the small set of SVG primitives used for the final rendering (e.g. only <rect>). Recently, some commercial and free softwares have been developed using some ad-hoc solution to the "raster to SVG" problem. Among other: Autotrace ([2]), Kvec ([8]), Vector Eye ([14]). Almost all software are devoted to SVG rendering of graphic images (e.g. clip art), showing in such case good performances but also several perceptual drawbacks when applied to digital pictures acquired by consumer devices.

2. EXPERIMENTS & CONCLUSIONS

A large set of experiments have been performed in order to evaluate the capabilities of the various techniques and comparing results between them. According to our results Vector Eye [14] has the best performances among commercial software, although only advanced solution ([4], [5]) are able to obtain high quality rendering. This is particularly true also considering for comparison both PSNR and compression ratio between original raster images and final SVG file (also gzipped). Figure 1 shows an image that has been vectorized with three advanced methods and with Vector Eye (i.e. magnified at 400%). Major details, on line demo and results can be found at the following web address: http://svg.dmi.unict.it/.

3. REFERENCES

[1] B. Antoniou, L. Tsoulos, *Converting Raster Images to XML and SVG*. In Proc. of SVGOpen, 2004

[2] Autotrace, Convert Bitmaps to Vector Graphics http://autotrace.sourceforge.net/, 2004

[3] S. Battiato, G. Gallo, G. Messina, *SVG Rendering of Real Images Using Data Dependent Triangulation*. In Proc. of ACM/SCCG2004, 2004

[4] S. Battiato, A. Costanzo, G. Di Blasi, G. Gallo, S. Nicotra, *SVG Rendering by Watershed Decomposition*, In Proc. of SPIE EI - Internet Imaging VI - Vol.5670.3, 2005

[5] S. Battiato, G. Barbera, G. Di Blasi, G. Gallo, G. Messina, *Advanced SVG Triangulation Polygonalization of Digital Images*, In Proc. of SPIE EI - Internet Imaging VI, Vol.5670.1, 2005

[6] D. Duce, I. Herman, B. Hopgood, *Web 2D Graphics File Format*. Computer Graphics *forum* - Vol.21(1), pp.43-64, 2002

[7] N. Dyn, D. Levin, S. Rippa, *Data Dependent Triangulation for Piecewise Linear Interpolation*. IMAJ. Numerical Analysis, Vol.10, 1990

[8] KVec, Raster Vector Conversion: http://www.kvec.de, 2004

[9] S. Lee, *Wavelet-Based Multiresolution Surface Approximation from Height Fields*, Ph.D. Thesis, Virginia State University, 2002

[10] A. Quint, *Scalable Vector Graphics*. IEEE Multimedia - Vol. 3, pp. 99-101, 2003

[11] D. Su, P. Willis, *Demosaicing of Color Images Using Pixel Level DDT*. In Proc. of IEEE Theory and Practice of Comp. Graph., pp. 16-23, 2003

[12] Scalable Vector Graphics – XML Graphics for the Web – http://www.w3c.org/Graphics/SVG, 2004

[13] SVG Open Conf. and Exhibition, Conf. on Scalable Vector Graphics, http://www.svgopen.com, 2004

[14] Vector Eye – Raster to Vector Converter - http://www.siame.com/index.html, 2004

[15] L. Vincent, O. Soille, *Watersheds in Digital Spaces: an Efficient Algorithm Based on Immersion Simulations*. IEEE Trans. on PAMI, Vol.13(6), pp. 583-598, 1991

An Architecture To Adapt Scalable Virtual Characters

Uwe Berner
TU Darmstadt, Interactive
Graphics Systems Group (GRIS)
Fraunhoferstr.5
64283 Darmstadt, Germany
uberner@gris.informatik.tu-darmstadt.de

Thomas Rieger
TU Darmstadt, Interactive
Graphics Systems Group (GRIS)
Fraunhoferstr.5
64283 Darmstadt, Germany
rieger@gris.informatik.tu-darmstadt.de

ABSTRACT

Virtual humans and avatars are enhancing a lot of current computer applications. This leads to the usage of complex characters for presentations, e-learning tasks and virtual reality applications. Often it might be suitable to have the possibilty of scaling the character to avoid performace problems or to fit the personal needs of the user. An universal architecture to control the scaling of the character is presented. It is based on an Adaptation Engine and different scalability modules which can be configured to the preferences of the user and the graphical system.

Keywords

Virtual Humans, Avatars, Scalability, Architecture

1. INTRODUCTION

Virtual humans and avatars are more and more present in current computer applications. This development includes complex 3D avatars for presentations, e-learning tasks and virtual reality applications. Neverthenless there are an emerging number of avatars guiding the user of small devices like PDAs and smartphones. In this context it could be useful to have the ability to scale the graphical representation of the character to avoid performance problems related to the used hardware, the complexity of the application or to fit the personal needs of the user. This could be especially useful on mobile devices. With this ability there is a wide range of avatar applications even on small computers and with complex and numerous running programms. In this paper we introduce an universal architecture to control the scalability of the avatars regarding different preferences of available scalability modules. With this system it is possible to change the scalability of the avatar when starting the system and during runtime.

WSCG 2005 POSTERS proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. RELATED WORK

There are a lot of systems dealing with virtual humans in different application context. Normally there is no adaptability of the graphical representation on the running system, if you need a less complex model you make it by loading a different one. Even you can not change other animation capabilities.

Neverthenless there are a some implementations dealing with the scalability of the graphical representation of the virtual humans during runtime. Here should be mentioned the adaptation of face and body animation in some MIRALab publications [Seo00],[Gia03] regarding different MPEG standards and the work of Breton et al. in [Bre01], [Bre02] and [Bre03] dealing with a FaceEngine to adapt an avatar on small devices. Especially Breton affiliated with the France Telecom has the explicit goal to run avatars on small devices, namely on smartphones.

3. ADAPTATION ENGINE

The Adaptation Engine is inspired from the necessity to scale our existing avatar systems to be used on smaller, less performant systems and mobile devices. For this we are developing a flexible system which can be used with different avatar systems incorporating different scalability modules.

One inspiration is the work of Breton et al. Scalability is controlled through scalability modules. Each module is connected with one or several animation modules or LOD-management. The scalability system is organized around a sorted list of scalability modules. The adaptation process starts with the highest module. When the execution process of the module found out that it is not enough performance, the next module is executed and so on. On the contrary, if there is remaining power, the system can go backward. Scalability modules that act on animations only have to decide if they switch them on or off. But scalability modules that act on LOD-management adjust dynamically the resolution of the meshes.

Our idea is to generalize the concept of Breton and additionally make it applicable to different avatar animation systems. The user should decide in which way the avatar is scaled, for this the capabilities of the system and the preferences of the user are important factors. We provide:

- Flexible execution of the scalability modules
- Connectibility with different avatar systems

The scalability modules are grouped in different levels, see Figure 1. The modules in one level are modified at the same time. Is the scalability of one level exhausted then the next level is reached, this works in both directions.

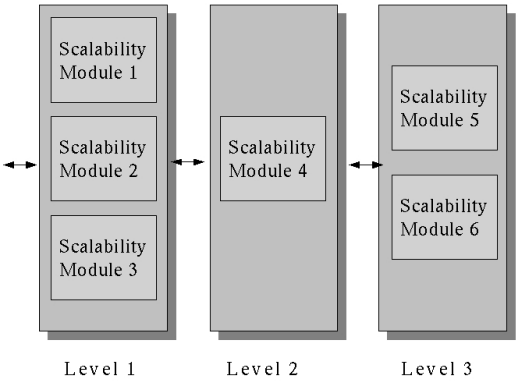


Figure 1 Levels of the Adaptation Engine.

The different modules can configurated with individual parameters like scalability type (linear or on/off), stepwidth, boundaries, startvalue, starttype (static/dynamic) and connection description. This gives the user the possibility to configure each scalability module depending on the requirements of an application.

With the described configuration capabilities scalability modules with different features can be connected with the Adaptation Engine. To connect an existing system including the scalability modules with the Adaptation Engine each module must have a stub for the communication. Another module called Performance Control is used the give the Adaptation Engine information of the current performance condition of the system. The configuration capabilities can be used to fit the Adaptation Engine to the existing scalability facilities of the avatar system and the preferences of a specific application. For the aquisition of more experience about the scalability of avatars we implemented two scalability modules with the basis of a Java3D avatar animation system:

- Number of Morph Targets
- Reduced Mesh

4. CONCLUSION

Starting with our work on avatar systems including scalability facilities and inspired from the work of Breton we introduced an architecture to adapt scalable virtual characters. This architecture includes an Adaptation Engine and scalability and performance modules with stubs. The adaptation Engine is flexible to configure and can be used with different avatar systems. Due to the configuration capabilities it can be adapted to the different needs and preferences of the system, the user and the current application context.

5. REFERENCES

[Bre01] G. Breton A 3D Facial Animation Engine for Real Time Applications, Web 3D Symposium (Paderborn, Germany 2001).

[Bre02] G. Breton, C. Bouville and D. Pelé, FaceEngine: Animation de Visages 3d parlants pour nouveaux IHM et services de télécommunications, PhD-Thesis (Rennes, France 2002).

[Bre03] G. Breton, Scalable Facial Animation, 3D Modelling 2003 (Paris, 2003).

[Gia03] T. Di Giacomo, C. Joslin, S. Garchery, N. Magnenat-Thalmann, Adaptation of Facial and Body Animation for MPEG-based Architectures, IEEE International Conference on CyberWorld (CW'03) (Singapore, 2003).

[Seo00] H. Seo and N. Magnenat-Thalmann, LOD Management on Animating Face Models, Proceedings of IEEE on Virtual Reality 2000 (New Jersey, 2000).

Texturing of Multi-Resolution Meshes with Basis Meshes

Hermann Birkholz
Scientific Assistant
IEF, University of Rostock
Albert Einstein Str. 21
18059, Rostock, Germany
hb01@informatik.uni-rostock.de

ABSTRACT

A well known problem in the field of multi-resolution meshes is attribute preservation. To simplify mesh geometry while surface attributes, like texture information, remain stable, the textures have to cover several parameterized triangles (mesh patches). In such cases we have to simplify the surface parameterization along with the geometrical simplification. To minimize visible deviation while simplifying, we need a suitable parameterization and must avoid drastic changes along the patch borders.

In this paper we present an algorithm that creates surface patches for multi-resolution meshes. These patches are parameterized to share textures and normal-maps for all possible mesh approximations. To create the patches, we simplify the mesh to a low resolution triangle mesh (basis mesh) whose triangle structure is projected to the original surface in an refinement step. The projected basis triangles are used to build the surface patches. These patches are finally parameterized with shape-preserving weights.

Keywords

surface parametrization, texture, meshes, multi-resolution.

1. Introduction

In order to increase the visible details of polygonal meshes, 2-d texture maps and normal-maps are used. These maps are normally applied to single triangles, mesh patches or the triangle mesh. To apply texture and normal-maps to triangle meshes, we need to parameterize the surface. Whenever several connected triangles of the mesh share the same texture, we have to map them into the 2-d space. We get their mapped coordinates and thus a local parameterization. When all triangles of a mesh are mapped into the same parameter space, we get a global parameterization. Our aim is to divide the mesh surface into triangular parameterized patches which are suitable for multi-resolution meshes.

POSTERS proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. Related Work

The creation of patch-based parameterizations for meshes has been investigated by a number of researchers. In [1] a hierarchical simplification was used to map the original surface vertices to a simplified version of the mesh. During the simplification the mapping process can cause triangle flips in the parameter space which must be treated separately. The resulting parameter spaces must finally be smoothed to obtain good results.

Another approach [2] creates so called "charts" from a mesh surface with an algorithm which iteratively merges two charts until a cost function stops the process. The borders of the resulting polygonal charts must be smoothed to be useable for multi-resolution meshes, but still limits the obtainable simplification ratio. The parameterizations of the final charts are optimized by means of the triangle stretch. This results in angle distortion for large patches and limits its usage with multi-resolution meshes.

Methods for global parameterizations where presented in [3,4,5]. But they are either limited to meshes of genus 0 [3] or limit the multi-resolution hierarchy with uneven mesh cuts [4,5].

3. Basis Domain Creation

In this section we will describe the creation of mesh patches, which can easily be parameterized with known techniques [6]. First of all we create an error based half-edge collapse [7] simplification hierarchy from the original mesh which leads to the basis domain mesh (coarsest level). The collapses are only allowed when the changed triangles can be projected into the exponential mapping of the former vertex star, without normal flips. For our tests we limited the number of triangles for the coarsest level but other stop conditions are possible too. During the mesh refinement, we iteratively map the domain mesh borders to the refined mesh. Due to the local influence of a vertex split (single vertex star) only few triangles have to be taken into account. Figure 1 shows the local mapping process.

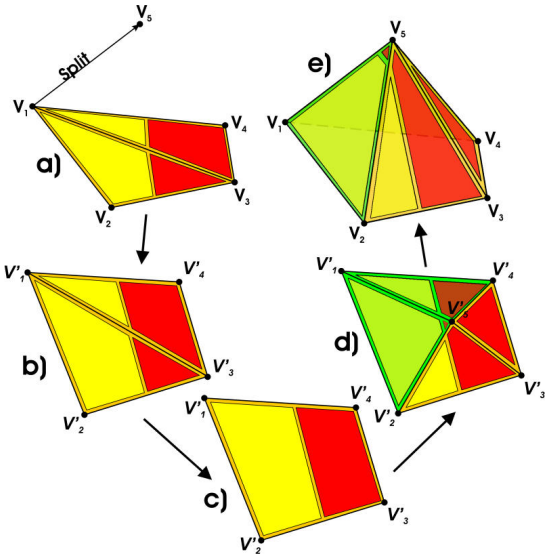


Figure 1: local split process

Fig.1a shows a vertex split, where $\mathbf{v_1}$ is split to a copy of itself and $\mathbf{v_5}$. The basis mesh structure mapped to the triangles ($\mathbf{v'_1}$, $\mathbf{v_2}$, $\mathbf{v_3}$) and ($\mathbf{v_3}$, $\mathbf{v_4}$, $\mathbf{v_1}$) is colored yellow and red. In the first step the border vertices are mapped to a plane with exponential mapping around $\mathbf{v_5}$ (Fig.1b). Then the basis domain edges are joined within the resulting polygon (1c). And finally we cut these connected base edges against the new triangles (1d) and store them in these. After the refinement, we triangulate the original mesh along the edges of the domain mesh to create smooth domain borders. Finally the domain borders and the inner points are parameterized to the domain triangles with the shape-preserving method [8].

The obtained parameterized mesh can now be simplified again to create the final simplification hierarchy. In this case we only allow edge collapses within the domain or along the domain borders, which must also be applied to the parameter space. The result is a parameterized multi-resolution mesh.

4. Results

The algorithm in Section 3 has been applied to several meshes to prove it. Every domain mesh could be mapped to the original mesh without problems and produced smooth parameterizations. Together with these parameter spaces we produced textured multi-resolution models. Figure 2 shows the parameterized "Stanford Bunny" in different approximations. The checker-box texture shows the smoothness of the parameterization, while the colors indicate the different domain triangles.

Figure 2: Bunny with 16000(u.l), 2000 (u.r), 500 (b.l.) and 50 (b.r.) triangles

5. References

[1] A. W. F. Lee, MAPS: Multiresolution adaptive parameterization of Surfaces, CG, 1998

[2] P. V. Sander, Texture mapping progressive meshes, CG and Interactive Techniques, 2001.

[3] C. Gotsman, Fundamentals of spherical parameterization for 3d meshes, SIGGRAPH, 2003

[4] N. Ray, Hierarchical least squares conformal maps, PG'03, 2003

[5] B. Levy, Least squares conformal maps for automatic texture atlas generation, CG and Interactive Techniques, 2001.

[6] K. Horman, Theory and Applications of Parameterizing Triangulations, PhD thesis, 2001

[7] L. Knobbelt, Interactive multi-resolution modelling on arbitary meshes, CG and Interactive Techniques, 1998

[8] M. S. Floater, Parameterization and smooth approximation of surface triangulations, CAGD, 1997

Locating sensors into a 3D polyhedral environment

Andrea Bottino
DAUIN – Politecnico di Torino
Corso Duca degli Abruzzi, 24
10128, Torino, Italy
andrea.bottino@polito.it

Aldo Laurentini
DAUIN – Politecnico di Torino
Corso Duca degli Abruzzi, 24
10128, Torino, Italy
aldo.laurentini@polito.it

ABSTRACT

In this paper we present an optimal 3D sensor location algorithms that can locate sensors into a polyhedral environment that are able to see the features of the objects in their entirety. Limitations due to real sensors can be easily taken into account. The algorithm has been implemented, and examples are also given.

Keywords

Inspection, surveillance, sensor planning, sensor positioning, art gallery problem, aspect graph, integer surface covering.

1. INTRODUCTION

Sensor planning is an important research area in computer vision. It consists of automatically computing sensor positions or trajectories given a task to perform, the sensor features and a model of the environment. Sensor panning problems require considering a number of constraints, first of all the visibility constraint. To this effect, the sensor is usually modeled as a point, and referred to as a "viewpoint". A feature of an object is said to be visible from the viewpoint if any segment joining a point of the feature and the viewpoint does not intersects the environment or the object itself and lies inside the frustum. In this paper we will deal with the problem of finding and locating the minimum number of sensors able to see at the same time all points of the surface of an object. The sensors are supposed to be omni directional. The major contribution of this paper is to present a 3D algorithm for finding a set of zones where a minimal set of viewpoints, able to observe the entire surface of the object, can be independently located. The algorithm works for multiply connected and unconnected general polyhedra, and can locate viewpoints able to observe the interior or the exterior surfaces of the polyhedra. For finding an optimal solution the view space needs not to be discretized, the only restriction being that each face must be completely observed by at least one viewpoint. It is also worth observing that, if the faces are subdivided into smaller areas, the solution provided by the algorithm converges towards the optimal solution of the unrestricted problem.

2. THE 3D ALGORITHM

The algorithm is an extension in 3D of the algorithm presented in [Bot04]. The outline of the 3D sensor positioning algorithm is the following:

Step 1 - Compute a partition Π of the viewing space into regions Z_i such that: a) the same set $\mathbf{F}_i = (F_p, F_q, ...F_t)$ of faces is completely visible from all points of $Z_i \; \forall$ i; b) the Z_i are maximum regions, i.e. $\mathbf{F}_i \neq \mathbf{F}_k$ for contiguous regions. The list of the visible faces must be associated to each region of Π.

Step 2 - Select the *dominant and essential regions* of Π. A region Z_i is defined to be dominant if there is no other region Z_k of the partition such that $\mathbf{F}_i \subset \mathbf{F}_k$. A region is *essentials* if it covers some face not covered by the other dominant zones (and must be selected).

Step 3 - Select an optimal (or minimum) solution. A minimum solution consists of a set of dominant regions $\mathbf{S}_j = (Z_{k1}, Z_{k2}, ..., Z_{kn})$ which covers all the faces with the minimum number of members.

The environment is assumed to consist of simple polyhedrons. Regarding the complexity, step 1 and 2 can be performed in polynomial time; step 3 requires, in the worst case, exponential time. However, polynomial time can be achieved using greedy heuristics. Partition Π is built by means of a particular set of surfaces, called the *active visibility surfaces*.

A *visibility surface* (VS) relative to a face divides the space into areas where the surface is partially or totally hidden. A VS is an half-open planar surface starting at one of the edges or at a vertex of a face, lying in the half space opposite to the inner side of the face. Also, each potential VS has a positive side, which is the side closer to the originating face. The angle between the normal of this surface and the normal of the originating face is in the range $[0,\pi]$. Examples can be seen in Fig. 1, where the arrows mark the positive side of the VSs.

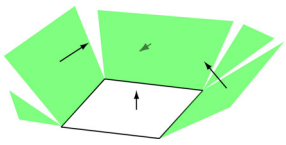


Fig. 1: example of VSs

Each VS can have an *active* and an *inactive* part. Only the active VSs are the effective boundaries of the visibility region of the corresponding surface. A VS is active when: 1) the angle with the normal of the originating face is 0 and the surface is not entering the object in the proximity of the originating face (VS of type I); 2) it is tangent to another part of the object (or to another object) and in the neighborhood of this part, the inner side of the object lies on the negative side of the potential VS (that is, the VE surfaces defined in [Gig91]). Those surfaces are defined by a vertex of the object and an edge of the face (VS of type II) or by an edge of the object and a vertex of the face (VS of type III). In both cases, the active part of the VS starts at the intersection point. Examples can be seen in Fig. 2. Each active VS has a cross operator `^j` associated, where `j` indicates the surface originating the VS. The operator has a direction, pointing to the area where the face is visible (see Fig. 2). In the

following we will use a result found in [Tar96], that is: if the face is convex (and simply connected), its visibility region is connected. This property allows pruning radically the initial set of potential VSs of a face. Therefore any concave face will be split into convex parts.

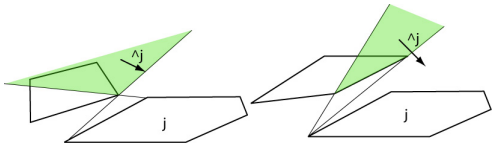


Fig. 2: VS of type II (left) and type III (right)

Finding the active part of a VS

For each initial VS, we must identify the part which is indeed active. In order to construct the active VSs of type I, we must find the regions of the plane P of a face F from where its 2D silhouette is completely visible. Forcing F to be convex, its 2D silhouette from a viewpoint V corresponds to the list of edges of F totally visible from the viewpoint. The active VS of type I can be constructed in the following way:

1. find on P the polygons corresponding to the intersection of the objects with P; let S, the initial active VS, be the union of all the regions in P where the 2D silhouette of F is completely visible;
2. for each edge, define as positive the side of the edge pointing to the internal part of the face; for each edge of the face closed by another face, or by another object intersecting the plane of the face, let H be the half plane in P bounded by the line containing the edge and corresponding to the positive side of the edge. Then S = S∩H (see Fig. 3)

Consider Fig. 3 where a face F and its active VS of type I are shown; edges e_1 and e_2 are closed by other objects.

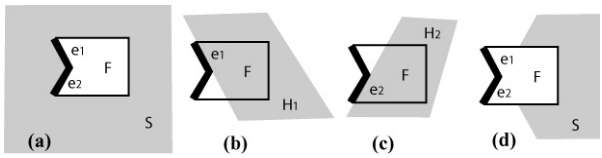


Fig. 3: edges e_1 and e_2 are closed by other objects

The initial active VS on P can be evaluated using a modified version of the 2D region labeling algorithm of [Bot04].

The active part of a VS of type II (III) can be found determining the parts of the initial unbounded triangular (trapezoidal) surface where the originating edge is entirely visible (see Fig. 4).

Additional rules for potential VS to be active

For a potential VS of type II or III, its orientation must be such that the plane containing the surface does not enter the inner side of F. For example, the VS of Fig. 5(a) is inactive.

Second, when the active parts of two VS relative to elements of the same face intersect somewhere, they both stop at their intersection (see Fig. 5(b)). The part of the VS that falls on the negative side of another VS becomes inactive.

Finally, consider a set of VS of type III (type II) insisting on the same edge e (vertex v). If one of the VS is found to be on the negative side of another VS, then it is inactive. See for instance Fig. 6(a) and (b).

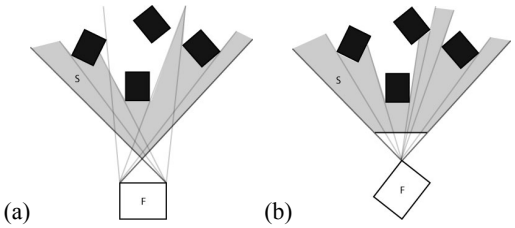


Fig. 4: active part of a VS of type II (a) and III (b)

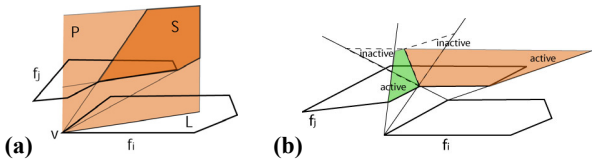


Fig. 5: inactive VS (a); partly inactive VS (b)

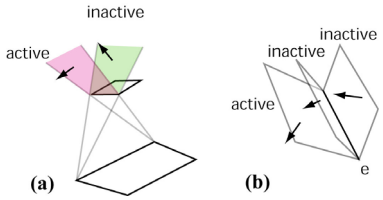


Fig. 6: only the most external surface is active

3. COMPUTING PARTITION Π

The outline of the algoritm for computing Π and labeling its regions, is:

1. find all the active VSs and the associate operator ^
2. intersect all the active VSs and subdivide the space into regions
3. select one region and compute the set of visibile faces V(f1,…,fn) for that zone
4. visit all the regions and compute their set of visible faces with the following rules:
   a. when crossing a boundary between R1 and R2 in the direction of the operator ^, the operand (j) is added to the set of visible faces of R2
   b. when crossing a boundary between R1 and R2 in the opposite direction of the operator ^, the operand (j) is removed from the set of visible faces of R2

4. CONCLUSIONS

In this paper a method for positioning a minimum number of sensors into a 3D polyhedral environment has been presented for some sample cases. The approach has been implemented and results have been presented. Future work will be focused on extending the algorithm in order to consider the general case of face covering, and not only its integer face covering restriction.

5. REFERENCES

[Bot04] A. Bottino, A. Laurentini "Optimal positioning of sensors in 2D". Proc. 9th Iberoamerican Congress on Pattern Recognition, Puebla (Mexico), October 26-29, 2004

[Gig91] Gigus, J. Canny, R. Seidel, “Efficiently computing and representing the aspect graphs of polyhedral objects”, IEEE Trans. PAMI, Vol. 13, no. 6, pp. 542-551, June 1991

[Tar96] K. Tarabanis, R.Y. Tsai, A. Kaul, “Computing occlusion-free viewpoint”, IEEE Trans. PAMI, Vol. 18, no. 3, pp. 279-292, march 1996

Identification of MPEG-4 FDP Patterns in Human Faces using Data-Mining Techniques

Britos, P.[1] pbritos@itba.edu.ar
Abásolo, M.[2 3] mjabasolo@uib.es
García-Martínez, R[1] rgm@itba.edu.ar
Perales, F.[2] paco.perales@uib.es

[1]Technology Centre of Buenos Aires (ITBA)
Centro de Ing. de Software e Ing. del Conocimiento.
Av. Madero 399
(1106) Ciudad de Bs.As. Argentina

[2]University of the Balearic Islands
Graphics and Computer Vision Unit.(UGiV)
Ed.A.Turmeda.Ctra.Valldemossa km7.5
(07122) Palma, Baleares, Spain

[3]National University of La Plata.
Department of Informatics. III-LIDI Institute
Calle 50 y Calle 115. 1er Piso.
(1900) La Plata. Bs.As., Argentina

ABSTRACT

In this paper we try to induce rules that describe patterns in human faces. We apply two different data-mining algorithms, C4.5 and C5.0, in a database of faces parameters in the MPEG4 FDP (Face Definition Parameters) form. Also we modify the database in two different ways before applying the algorithms: variable discretization of some fields; and selection of the main clusters with Self-Organizing Maps.

Keywords

Data-Mining, Self-Organizing Maps, C4.5, C5.0, MPEG-4 Face Definition Parameters, FDP

1. INTRODUCTION

MPEG-4 [ISO98a] is an ISO/IEC standard developed by MPEG (Moving Picture Experts Group), which defines ways to represent and compress audio, video and 2D/3D graphics objects. Particularly MPEG-4 defines 84 feature points called *Face Definition Parameters* (*FDPs)* to parameterise a face. FDPs are used to personalize a generic face model to a particular face.

In our work we have a database of faces that are described by distances between different *MPEG-4 (FDP)*. The main purpose of this work is to induce rules that describe patterns in the human faces, that means relations between different dimensions of the faces from the database (i.e. mouth width, eyebrown width, etc.)

Data mining is all about extracting patterns from a warehoused data. C4.5 is an automatic learning algorithm developed by Quinlan [Qui92a] that has been used for classifying examples. The classifiers are expressed as decision trees or sets of if-then rules forms that are generally easier to understand than neural networks. C5.0 is an improvement of C4.5 algorithm [Rul03a].

In this work we compare the results of applying C4.5 and C5.0 to the database of MPEG-4 FDPs. Rules have the "if-then" form and refer relations between different dimensions of the face, i.e. "if eye distance is bigger than… then mouth is…width".

Self Organizing Maps (SOM) [Koh01a] is the most popular artificial neural network algorithm in the unsupervised learning category. In this work we use SOM for clustering the records in the database of faces before applying C4.5 and C5.0 algorithms.

Conference proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2.EXPERIMENTATION

We have a database of faces where each record has the 19 fields shown in first column of Table 1. Except the first four fields -sex, human race, height and weight- the rest of the fields are segments of the face from one MPEG-4 FDP to another.

We apply the C4.5 and C5.0 algorithms to: the entire database (case a) and the main cluster obtained with Self Organizing Maps (case b).

Field	Commentaries
Sex	male/female
Human race	asiatic, european, etc.
Height	person's height in meters
Weight	person's weight in kg.
Face height (FH)	FPD 11.1 to FPD 2.1
Face width (FW)	FPD 10.10 to FPD 10.9
Right eye width (REW)	FPD 3.12 to FPD 3.11
Right eye height (REH)	FPD 3.14 to FPD 3.10
Right iris diameter (RID)	FPD 3.2 to FPD 3.4
Left eye width (LEW)	FPD 3.8 to FPD 3.7
Left eye height (LEH)	FPD 3.13 to FPD 3.9
Left iris diameter (LID)	FPD 3.1 to FPD 3.3
Right eyebrown (RE)	FPD 4.2 to FPD 4.6
Left eyebrown (LE)	FPD 4.1 to FPD 4.5
Nose height (NH)	FPD 9.6 to FPD 9.2
Nose angle (NA)	Angle between NH and FPD 9.2 to FPD 9.1
Nose Tip (NT)	FPD 9.3 to FPD 9.15
Mouth height (MH)	FPD 8.1 to FPD 8.2
Mouth width (MW)	FPD 8.4 to FPD 8.3

Table 1.Face record

These are the rules obtained from the entire database (case a) with the objective field "Sex":

Rules obtained with C4.5

IF weight < 63 kg. THEN sex = female

IF NA >= 81,57º THEN sex = female

IF weight >= 72 kg. THEN sex = male

IF weight < 72 kg. THEN sex = female

IF RID >= 23 mm THEN sex = male

Rules obtained with C5.0

IF weight < 70 kg. AND LE <=79mm THEN sex = female

IF weight <= 62 kg. THEN sex = female

IF weight >= 62 kg. AND LE >79mm THEN sex = male

IF weight > 70 kg. THEN sex = male

After applying Self Organizing Maps for clustering the database records according its similarities we obtain four groups of records distributed as follows: 2% in cluster 1, 10% in cluster 2, 32% in cluster 3 and 56% in cluster 4. These are the rules obtained from the main cluster (case b) with the objective field "Sex":

Rules obtained with C4.5

IF weight < 73 kg. THEN sex = female

IF height >=1.73m. THEN sex = male

Rules obtained with C5.0

Objective Field:sex

IF height <=1.69m. THEN sex = female

IF REH >54mm. THEN sex = female

IF height >1.69m. AND REH <=54mm THEN sex = male

We also change the objective field to "FH" and "FW". Since these are continuos fields a variable discretization proccess allows to use them as objetive field in the rules.

Table 2 shows for every test the number of records that can not be analysed. As this percentage dismiss the rules are more precise.

	Entire database		Main Cluster	
	C4.5	C5.0	C4.5	C5.0
No discretization	39%	0%	18%	2,9%
Discretization	58%	17.7%	43%	16.7 %

Table 2. Percentage of non-analyzed records

C5.0 vs. C4.5

- The number of records that can not be analyzed dismisses considerably by applying C5.0 instead of C4.5.
- The rules deduced with C5.0 are more complex (left part of the rule has a conjunction).

Cluster vs. entire database

- The number of records that can not be analyzed dismisses by analysing the main cluster instead of the whole database in almost all the cases.

3. CONCLUSIONS

In this paper we determine rules to identify patterns in human faces. We analyze a database of faces described by dimensions from MPEG-4 Face Definition Parameters. We compare C4.5 data-mining algorithm with a lightly different algorithm called C5.0 to conclude that the latter one is more precise to induce rules. Also we conclude that the rules a better if we use only the records of the main cluster selected by Self-Organizing Maps instead of using the entire database. The discretization of some fields allows using it as an objective of the rules.

4. REFERENCES

[ISO98a] Information Technology. Generic Coding of Audio Visual Objects. Part 2: Visual. ISO/IEC FDIS 14496-2 Final Draft International Standard, 1998

[Koh01a] Self-Organizing Maps. Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, ISBN 3-540-67921-9, ISSN 0720-678X, 1995, 1997, 2001.

[Qui92a] Quinlan, J.R. C4.5: Programms for Machine Learning (Morgan Kaufmann, San Mateo,CA, 1992.

[Rul03a] RuleQuest Research. Data Mining Tools See5 and C5.0 http://www.rulequest.com/see5-info.html, 2003

A New Hilbert Invariant For A Pattern Of Points Under Affine Transformation

C.H. Chan, Y.S. Hung and C.H. Leung
Department of Electrical and Electronic Engineering
The University of Hong Kong, Pokfulam Road, Hong Kong, P.R.China
{chchan, yshung, chleung}@eee.hku.hk

ABSTRACT

This paper presents a new Hilbert Invariant for a pattern of points in an image. The Hilbert invariant is invariant under any affine transformation (translation, scaling, rotation and shearing). It is constructed based on the Hilbert transform. Hilbert transform is originally used for generating the imaginary part from the real part of a continuous or discrete complex signal to recover the original one in signal processing. In this paper, the real part is a discrete signal formed by a sequence of coordinates of image points. The Hilbert transformed signal is then used to construct a relative and an absolute affine invariant.

Keywords

Affine transformation, Image points, Hilbert transform, Hilbert invariant

1. INTRODUCTION

The Hilbert transformation is a process performed on a real continuous time-domain signal $u(t)$ yielding a new signal $v(t)$ to generate an analytic complex-valued signal $\psi(t) = u(t) + jv(t)$. Given a time function $u(t)$, the Hilbert transform is [Han96a]:

$$H[s]=\frac{1}{\pi}\int_{-\infty}^{+\infty}\frac{u(t)}{s-t}dt; \quad -\infty<t<\infty \ldots(1)$$

The variable s here is a time variable, so the Hilbert transform of a time function is another time function.

The following section presents the derivation of Hilbert invariant based on the Hilbert transform of the coordinates of image points.

2. AFFINE HILBERT INVARIANT

Given a sequence $x_R[n]$ of finite length N taken to be the real part of a complex sequence. Then, the imaginary part $x_I[n]$ of finite length N is defined as the circular convolution of $x_R[n]$ with the impulse response $h[n]$ of the Hilbert transform [Joh99a]:

WSCG 2005 POSTERS proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

$$x_I[k]=x_R[k]\otimes h[k]=\sum_{m=0}^{N-1}x_R[m]h[k-m], 0\le k\le N-1 \ldots(2)$$

Now let $x_R[k]$ and $y_R[k]$ be two sequences of finite length N which are the x-coordinates and y-coordinates respectively of the ordered point set:

$$x_R[k]=[x_0,\ldots\ldots,x_{N-1}],\ y_R[k]=[y_0,\ldots\ldots,y_{N-1}]$$

Then $$x_I[k]=\sum_{m=0}^{N-1}x_R[m]h[k-m] \ldots (3)$$

$$y_I[k]=\sum_{m=0}^{N-1}y_R[m]h[k-m] \ldots (4)$$

Let $A=\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$ be an affine transformation and $x_R'[k]$ and $y_R'[k]$ be two sequences of finite length N which are the x-coordinates and y-coordinates respectively of the affine transformed ordered point set:

$$x_R'[k]=[x_0',\ldots\ldots,x_{N-1}'],\ y_R'[k]=[y_0',\ldots\ldots,y_{N-1}']$$

where $\begin{bmatrix} x_i' \\ y_i' \end{bmatrix}=\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix}\begin{bmatrix} x_i \\ y_i \end{bmatrix}+\begin{bmatrix} t_1 \\ t_2 \end{bmatrix}, \quad i=0,\ldots,N-1.$

$$\therefore x_I'[k]=\sum_{m=0}^{N-1}x_R'[m]h[k-m]$$
$$=\sum_{m=0}^{N-1}\left[r_{11}x_R[m]+r_{12}y_R[m]+t_1\right]h[k-m]$$
$$=r_{11}(x_I[k])+r_{12}(y_I[k])+t_1\sum_{m=0}^{N-1}h[k-m] \quad \ldots(5)$$

It can be shown that in both odd and even N cases:

$$\sum_{m=0}^{N-1} h[k-m] = 0 \quad \text{for } 0 \le k \le N-1 \quad ...(6)$$

Hence, substituting (6) into (5):

$$x_I'[k] = r_{11}(x_I[k]) + r_{12}(y_I[k]) \quad ...(7)$$

Similarly, $y_I'[k] = r_{21}(x_I[k]) + r_{22}(y_I[k])$ … (8)

Combining (7) & (8), $\begin{bmatrix} x_I'[k] \\ y_I'[k] \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x_I[k] \\ y_I[k] \end{bmatrix}$ ...(9)

Therefore, for $0 \le k_1, k_2 \le N-1$,

$$\begin{bmatrix} x_I'[k_1] & x_I'[k_2] \\ y_I'[k_1] & y_I'[k_2] \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x_I[k_1] & x_I[k_2] \\ y_I[k_1] & y_I[k_2] \end{bmatrix} \quad \text{… (10)}$$

Taking determinant on both sides:

$$\begin{vmatrix} x_I'[k_1] & x_I'[k_2] \\ y_I'[k_1] & y_I'[k_2] \end{vmatrix} = \alpha \begin{vmatrix} x_I[k_1] & x_I[k_2] \\ y_I[k_1] & y_I[k_2] \end{vmatrix} \quad ...(11), \quad \alpha = \begin{vmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{vmatrix}$$

Equation (11) is a relative affine invariant. An absolute affine invariant can be constructed by taking the ratio of two relative affine invariants:

$$HI = \frac{\begin{vmatrix} x_I'[k_1] & x_I'[k_2] \\ y_I'[k_1] & y_I'[k_2] \end{vmatrix}}{\begin{vmatrix} x_I'[k_3] & x_I'[k_4] \\ y_I'[k_3] & y_I'[k_4] \end{vmatrix}} = \frac{\begin{vmatrix} x_I[k_1] & x_I[k_2] \\ y_I[k_1] & y_I[k_2] \end{vmatrix}}{\begin{vmatrix} x_I[k_3] & x_I[k_4] \\ y_I[k_3] & y_I[k_4] \end{vmatrix}} \quad \text{…(12)}$$

for $0 \le k_1, k_2, k_3, k_4 \le N-1$, which is the Hilbert Invariant.

3. DISCUSSION

Given an input image of a set of ordered points of an oriented polygon extracted from the contour of an object after edge detection and thinning, we apply the Hilbert transform to them. As Hilbert invariant is invariant to the translation component of an affine transformation (see (7), (8)), there is no need to move the coordinate system to the area center. Let (x_i, y_i), $i = 0,\ldots,N\text{-}1$ be the coordinates of the N Hilbert transformed points, where $(x_0, y_0) = (x_N, y_N)$. We calculate the Hilbert invariant for each point along the polygon. Finally, a vector of invariants is constructed to represent the object for further processes such as recognition or matching.

Since area ratio is one of the commonly used affine invariants for matching, the noise performance of the area ratio is compared with the Hilbert invariant.

Given $0 \le k_1, k_2, k_3, k_4, k_5, k_6 \le N-1$, the area ratio Δ_{125}:Δ_{346} is defined as:

$$\frac{\begin{vmatrix} x_R'[k_1]-x_R'[k_5] & x_R'[k_2]-x_R'[k_5] \\ y_R'[k_1]-y_R'[k_5] & y_R'[k_2]-y_R'[k_5] \end{vmatrix}}{\begin{vmatrix} x_R'[k_3]-x_R'[k_6] & x_R'[k_4]-x_R'[k_6] \\ y_R'[k_3]-y_R'[k_6] & y_R'[k_4]-y_R'[k_6] \end{vmatrix}} = \frac{\begin{vmatrix} x_R[k_1]-x_R[k_5] & x_R[k_2]-x_R[k_5] \\ y_R[k_1]-y_R[k_5] & y_R[k_2]-y_R[k_5] \end{vmatrix}}{\begin{vmatrix} x_R[k_3]-x_R[k_6] & x_R[k_4]-x_R[k_6] \\ y_R[k_3]-y_R[k_6] & y_R[k_4]-y_R[k_6] \end{vmatrix}}$$

To have a better comparison, the Hilbert invariant defined in (12) is modified as follow:

$$\frac{\begin{vmatrix} x_I'[k_1]-x_I'[k_5] & x_I'[k_2]-x_I'[k_5] \\ y_I'[k_1]-y_I'[k_5] & y_I'[k_2]-y_I'[k_5] \end{vmatrix}}{\begin{vmatrix} x_I'[k_3]-x_I'[k_6] & x_I'[k_4]-x_I'[k_6] \\ y_I'[k_3]-y_I'[k_6] & y_I'[k_4]-y_I'[k_6] \end{vmatrix}} = \frac{\begin{vmatrix} x_I[k_1]-x_I[k_5] & x_I[k_2]-x_I[k_5] \\ y_I[k_1]-y_I[k_5] & y_I[k_2]-y_I[k_5] \end{vmatrix}}{\begin{vmatrix} x_I[k_3]-x_I[k_6] & x_I[k_4]-x_I[k_6] \\ y_I[k_3]-y_I[k_6] & y_I[k_4]-y_I[k_6] \end{vmatrix}}$$

Let $k_1 = k_3 = i$, $k_2 = i+1$, $k_4 = i+2$, $k_5 = k_6 = i+3$. Both invariants are then tested on 50 sets of 100 randomly generated points. Gaussian noise is added to each coordinate of the points. The standard deviation of noise is from 0.2 to 3 with 0.2 increments. The error percentage of an invariant for each point in each set is calculated and the average error percentage is found. The following figure shows the comparison result:

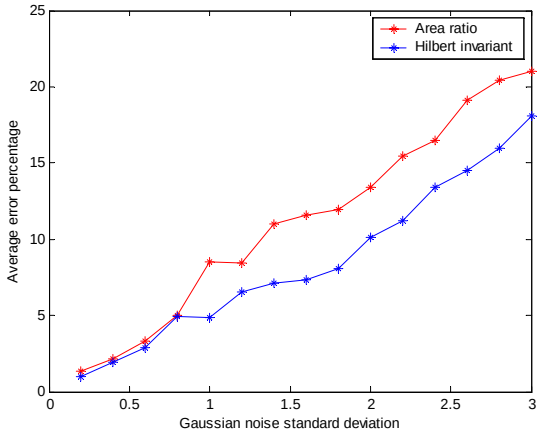


Figure 1. Comparison of robustness to noise.

From the figure, it can be seen that the performances of both invariants are similar in a low noise condition, but the Hilbert invariant performs better in a higher noise condition, so the Hilbert invariant is more robust to noise.

4. CONCLUSION

A new Hilbert Invariant for a pattern of points in an image is constructed based on the Hilbert transform. The Hilbert transformed sequence of coordinates of image points is then used to construct a relative and an absolute affine invariant.

5. ACKNOWLEDGMENTS

The work described in this paper was supported by a grant from the Research Grants Council of Hong Kong SAR, China (Project No.HKU7058/02E) and partially by CRCG of the University of Hong Kong.

6. REFERENCES

[Han96a] Stefan L. Hahn, “Hilbert transforms in signal processing”, Artech House publishers, Boston, London, 1996.

[Joh99a] Mathias Johansson, “The Hilbert transform”, master thesis, Mathematics/Applied Mathematics, Vaxjo University, 1999.

A Local Error Bound Approach to Simplifying Complex Geometric Models

John Cox
Staffordshire University
Beaconside, Stafford
United Kingdom ST18 ODG
j.s.cox@staffs.ac.uk

Claude C. Chibelushi
Staffordshire University
Beaconside, Stafford
United Kingdom ST18 ODG
c.c.chibelushi@staffs.ac.uk

ABSTRACT

This paper presents a new error bound simplification algorithm for complex geometric models. A lower polygon count approximation of the input model is generated by performing edge collapse operations. The collapse vertex is constrained to lie within a localised tolerance volume built around the edge collapse neighbourhood. This constraint ensures that all points on the simplified surface are within a user specified distance from the surface after the previous edge collapse.

Keywords

Simplification, level of detail, tolerance volume.

1. INTRODUCTION

This paper presents a new simplification algorithm for geometric models. The algorithm develops upon the simplification envelope approach proposed in [Coh95] and [Bre00], by creating an arbitrarily tight, localised tolerance volume built around the edge collapse neighbourhood. Constraining the collapse vertex to lie within this tolerance volume guarantees that all points on the simplified surface lie within a user specified distance from the surface after the previous edge collapse operation.

2. THE TOLERANCE VOLUME

The tolerance volume is built around the triangles making up the collapse neighbourhood (e.g. those triangles sharing either vertex of the collapsing edge) and represents a subset of collapse vertex positions that preserve a user specified bound on simplification error. Firstly, the convex kernel is created from the boundary edges (edges not sharing a vertex with the collapsing edge) within the collapse neighbourhood by constructing boundary planes lying orthogonal to its corresponding boundary triangle and passing through its boundary edge. The convex kernel of the collapse neighbourhood is then intersected with a set of prisms created for each triangle within the collapse neighbourhood. The prisms are formed by offsetting each triangle along its positive and negative surface normal by a distance equal to the error bound. The open sides of the prisms are capped with quadrilaterals. A typical prism is shown in Figure 1.

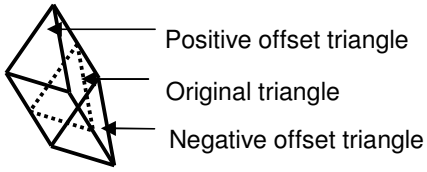


Figure 1. Prism used in the construction of the tolerance volume.

The set of prisms defined above may contain illegal vertex positions. A position is illegal if it results in any part of the simplified surface breaching the error bound. This problem is overcome by partitioning the prisms into legal and illegal sub-regions with partition sets.

2.1 Partition Sets

Partition sets are created for the offset triangles within each prism. There are three basic varieties of partition: those for boundary prisms (created for triangles sharing only one vertex with the collapsing edge), those for internal prisms (created for triangles sharing both vertices of the collapsing edge) and those where the boundary vertex used to construct the partition is external to the prism used in the construction of the partition.

2.1.1 Boundary Prisms

Partitions are created for the offset triangles within each boundary prism. The vertices belonging to the internal edges (e.g. those sharing one of the offset collapsing vertices) on each offset triangle are displaced along the vector formed between the opposite boundary vertex (on the surface triangle

WSCG 2005 POSTERS proceedings, ISBN 80-903100-8-7,
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

used to construct the prism) and the vertex to be displaced. The vertices need to be displaced by a large enough distance to ensure that the partition is capable of intersecting all prisms. The vertices of the internal edge together with the displaced vertices define a quadrilateral partition. Since there are two internal edges on an offset triangle within a boundary prism, two quadrilateral partitions are formed. The gap between the two partitions is capped to form a triangle. The offset triangle itself is added to the set making a total of four partitioning polygons. The complete partitioning set for a boundary prism is illustrated in Figure 2(a).

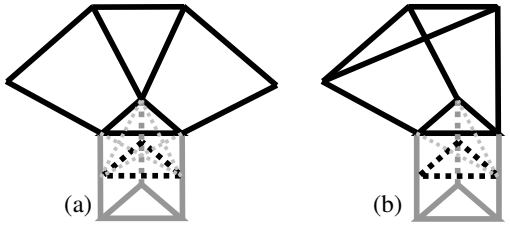


Figure 2. Partition set for a boundary (a) and internal (b) prism

2.1.2 Internal Prisms

The partition sets for internal prisms are formed in an analogous fashion to those of boundary prisms. The difference lies in the fact that the offset triangles within internal prisms have three internal edges. A partitioning set for an internal prism is illustrated in Figure 2 (b).

2.1.3 Exterior Partitions

Exterior partitions are created for both boundary and internal prisms in a similar fashion to those of internal prisms except that the boundary vertex lies within another prism. Partition sets are created for all boundary vertices lying on the inside (with respect to the prism) of the half-space formed from any internal edge on an offset triangle and the corresponding edge on the surface triangle used to construct the prism.

Each partition set represents a convex region defined by four half-spaces. The partition set divides space into a legal and an illegal region, with the prism used to construct the partition defined as being in the legal region. Each partition set is used to partition the prisms and the illegal regions are discarded. The remaining regions represent a sub-set of all legal collapse vertex positions.

2.2 Measuring Simplification Error

Surface approximation error is measured by projecting the simplified surface onto the original surface and calculating the maximum distance between corresponding points on the two surfaces. The simplified surface within each prism is projected by parallel orthographic projection onto the surface triangle used to construct the prism. The distance between the simplified surface and the original surface changes linearly within each prism. Hence, the maximum error will occur at a vertex on the simplified surface within a prism or at an intersection point between the simplified surface and the sides of a prism. An illustrative two-dimensional example is given in Figure 3 which shows 2D quadrilateral analogues to prisms. The target vertices and intersection points are highlighted with circles in the diagram. To calculate the maximum error, it is only necessary to project these target points onto the original surface within the appropriate prism (quadrilateral) and search for the maximum distance between corresponding points.

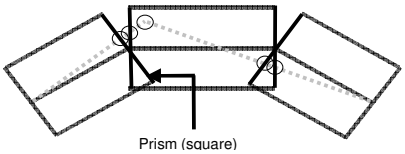


Figure 3. Illustrative example of a 2D edge collapse showing quadrilateral analogues of prisms and the simplified surface (dotted).

3 RESULTS

The local error bound algorithm has been implemented to perform all legal edge collapses within a user specified global error bound. The global error bound is set manually to some fraction of the diagonal of the bounding box of the model. The error bound used in the construction of tolerance volumes is set to a small fraction of the global error bound and all legal collapses are performed. The error bound is gradually increased until the desired triangle count is reached, or the error bound exceeds the user specified global error bound. The results have shown that the algorithm is capable of large scale polygon reduction while preserving important surface features.

4 REFERENCES

[Bre00] Bremer, P., Kreylos, O., Hamann, B., Wolter, F., Simplification of closed triangulated surfaces. M.L. Haungs, ed., Proc. of the 2000 UC Davis Student Workshop on Computing, TR CSE-2000-9, California, 2000.

[Coh95] Cohen, J., Varshney, V., Manocha, D., Turk, G., Weber, H., Agarwal, P., Brooks, F., Wright, W., Simplification envelopes. Proc. SIGGRAPH, pp. 119-128, 1995.

Fast Photomosaic

Gianpiero Di Blasi Maria Petralia

D.M.I. - University of Catania
Via A. Doria, 6
95125, Catania, Italy

gdiblasi@dmi.unict.it ankh76@virgilio.it

ABSTRACT

Photomosaic is a technique which transforms an input image into a rectangular grid of thumbnail images preserving the overall appearance. The typical photomosaic algorithm searches from a large database of images one picture that approximates a block of pixels in the main image. Since the quality of the output depends on the size of the database, it turns out that the bottleneck in each photomosaic algorithm is the searching process. In this paper we present a technique to speed-up this critical phase using the Antipole Tree Data Structure. This improvement allows the use of larger databases without requiring much longer processing time.

Keywords
Photomosaic, Antipole tree, non-photorealistic rendering, image processing and enhancement

1. INTRODUCTION

Photomosaic [Sil97] is a technique which transforms an input image into a rectangular grid of thumbnail images.
The bottleneck in each photomosaic algorithm is the search in a large database of images to find a best match. This search is usually sequential and the time required to perform this task is, hence, high. Some photomosaic algorithm relies on the off-line construction of a suitable data structure to speed-up the matching process. However the overall time required to create this structure could be excessive.
In this paper we propose a new method to speed-up the search process. This technique is based on the Antipole Tree Data Structure [Can04] and allows us to obtain impressive effects in an efficient manner. The Antipole Tree is suitable for searches over large record sets embedded into a metric space (X, d).
The rest of this paper is organized as follows: Section 2 introduces the Antipole Clustering Strategy, Section 3 explains our algorithm. In Section 4 we show the experimental results. Finally in Section 5 we suggest directions for future work and research.

POSTERS proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. THE ANTIPOLE CLUSTERING STRATEGY

The Antipole Clustering Strategy of bounded radius is a top-down procedure that starts with a given finite set of points X in a metric space. The first check is if there exists a pair of points in X such that their distance is longer than the radius. If this is the case, the set is partitioned by assigning each point of the splitting subset to the closest endpoint of the pair (A, B). Otherwise the splitting is not performed and the given subset is itself a cluster.
Once the data structure is built a suitable nearest neighbor algorithm can be designed. The search, starting from the root, proceeds by following the path in the tree, which guarantees to find the nearest cluster centroid pruning the impossible branches. A backtracking search explores the remaining branches of the tree to assure a correct answer.

3. THE PROPOSED ALGORITHM

The proposed algorithm can be divided into two steps: database acquisition and photomosaic creation.

3.1 Database Acquisition

The acquisition of the database of images is very simple: we partition each image of the database into 9 equal rectangles arranged in a 3x3 grid and compute the RGB mean values for each rectangle. This leads us to a vector x composed by 27 components (three RGB components for each rectangle). x is the feature vector of the image in the data structure. When all the images in the database have their own feature vector the Antipole clustering can be performed.

3.2 Photomosaic Creation

The photomosaic creation is very simple and easy to explain in few steps. First we subdivide the input image into a regular grid, then each cell of the grid into another 3x3 sub-grid. Second we compute the RGB mean values for each sub-cell of the sub-grid. This leads us to a vector x composed by 27 components (three RGB components for each sub-cell). x is the feature vector of the cell. After performing the best matching we resize the selected tile to fit and paint it over the cell.

4. EXPERIMENTAL RESULTS

In this Section we report some examples and quantitative results. All experiments have been carried out on a PC Athlon XP-M 1800+. To allow the reader to test directly the quality of our algorithm an applet is available at the URL [Dib04].

The examples in Figure 1 show the effectiveness of the proposed technique. Timing results (Table 1) show that our algorithm is fast enough to be used as a plug-in in a typical user-end software.

5. CONCLUSIONS & FUTURE WORK

In this paper we presented a new method to speed-up the creation of photomosaic images. Experimental results show the soundness of our algorithm.

There are several ways to improve the aesthetic of our results and several ideas started from this work:

1. extending photomosaic technique to other kind of mosaics as proposed in [Dib04b] and [Elb03];
2. the use of Antipole Tree or other data structures in other fields of non-photorealistic rendering to speed-up the rendering process, for example the JIM algorithm proposed in [Kim02];
3. extension of our method for photomosaic rendering of 3D surface is probably the most exciting direction of research.

6. REFERENCES

[Can04] Cantone D., Ferro A., Pulvirenti A., Reforgiato Recupero D., Shasha D. Antipole Tree indexing to support range search and K-nearest neighbor search in metric spaces. Accepted to IEEE Transactions on Knowledge and Data Engineering, 2004

[Dib04] Di Blasi G., Petralia M. The Photomosaic Creator applet.
www.dmi.unict.it/~gdiblasi/photomosaic/photomosaic.html
JGimp plug-in and Java application
www.dmi.unict.it/~gdiblasi/photomosaic/photomosaic.jar

[Dib04b] Di Blasi G., Gallo G. Artificial Mosaic. Submitted to The Visual Computer, 2004

[Elb03] Elber E, Wolberg G. Rendering Traditional Mosaics. The Visual Computer 19, pp 67-78, 2003

[Kim02] Kim J. and Pellacini F. Jigsaw Image Mosaics. In proceedings of SIGGRAPH2002, pp 657-664, 2002

[Sil97] Silvers R. and Hawley M. Photomosaics. Henry Holt, New York, 1997

Size	Total Mean Time (sec.)	Size	Total Mean Time (sec.)
320x240	5.980	800x600	19.058
640x480	16.044	1024x768	32.487

Table 1. Timing results

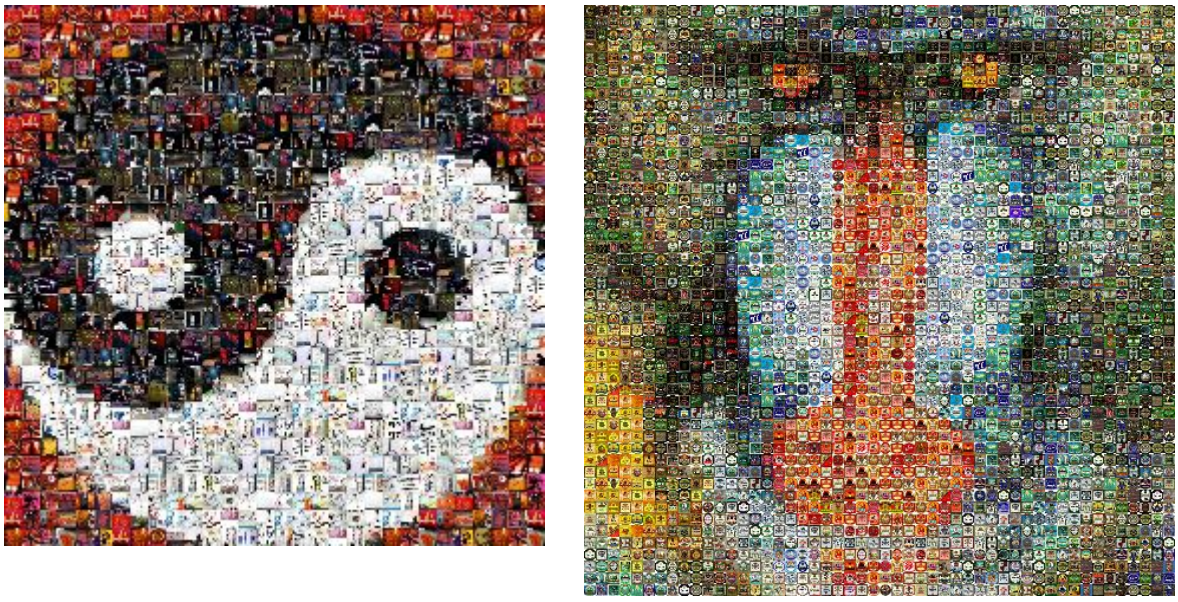


Figure 1. Some examples of our photomosaic algorithm

Segmentation of Colour Regions from Two-Colour Halftone Prints

Miroslav Fribert
University of Pardubice
Studentská 95
53210 Pardubice, Czech Republic
miroslav.fribert@upce.cz

Abstract

To evaluate print characteristics on the basis of scanned microsamples of printing sheets, it is necessary to carry out the segmentation of particular colour regions, printed with process inks. The accuracy of the evaluation of print characteristics strongly depends on the successful segmentation of the colour regions involved in a print sample. This paper describes the colour region segmentation based on the combination of thresholding and edge detection methods.

Keywords

Halftone print, print characteristics, colour, segmentation, edge detection.

1. Introduction

Colour publications are printed with using four process inks – yellow (Y), magenta (M), cyan (C) and black (K) - by various technologies of industrial press. The results of such printing are halftone dots of various sizes printed with various process inks. We can measure and evaluate the basic characteristics, which determine quality of the print, from the halftoned two-colour print samples.

The print characteristics used to examine print quality which are measured in the solid and halftoned areas are:

Optical density of the ink layer which determines quality of ink printing on the paper, *Dot area* which determines enlargement of print dots and *Ink trapping* which determines quality of ink printing to ink layer printed sooner) .

POSTERS proceedengs ISBN 80-903100-8-7
WSCG '2005, January 31-February 4, 2005
Plzen, Czech Republic

These parameters can be evaluated also from two-colour print samples scanned with a CCD camera and processed with image analyses methods. The usual practice is to perform the segmentation of an individual process colour and their overprint regions from the print sample, and then evaluate the print characteristics. In this case these are optical density inside the segmented regions, ink trapping gained from these densities and geometrical dot area. Two-colour print samples are usable for this purpose, because of the possibility of evaluation of all three characteristics from one sample.

The crucial point for the successful evaluation of print characteristics is the exact segmentation of colour regions in the sample. There are many methods, how to accurate results of segmentation [1]. The combination of thresholding grey-level image gained from original colour image by separation process, and edge detection in the original colour image was used for this purpose.

2. Combined segmentation method

This method is based on using colour separation of the sample into a grey level image, in which the region of the separated colour has the minimum brightness [5].

On the fig.1 see separations of blue and yellow regions gained by separation process.

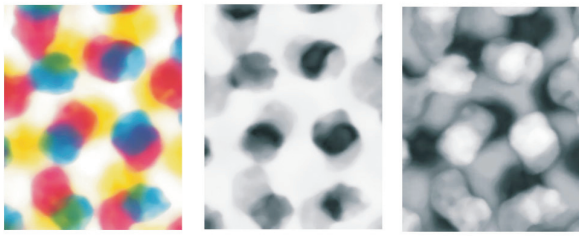


Fig.1 Separations of blue and yellow

A method of combination the thresholding and edge detection is based on an assumption that the border of the particular colour region lies in the middle of the edge of colour change-over from the first to the second colour. We can calculate significant edges of the colour region, which are to be segmented, with help of some edge detector.

The co-ordinates of edge pixels are the pointers to grey-level separated image and determine a set of threshold values. The average of these thresholds is the optimal threshold for the segmentation of the colour region.

3. Edge detection of colour regions

The edge detection method used in this work is based on the colour differences appearing in the pixel neighbourhood. The statistical parameter variance of the colour values in this neighbourhood indicates the edge size.

The variance of the component *hue* from HSB colour space as the edge detector was used in this work. The main advantage of this approach is smoothing of the colour image simultaneously with the edge detection. This is because *saturation* and *brightness* components include the major portion of the image noise in halftoned printing images.

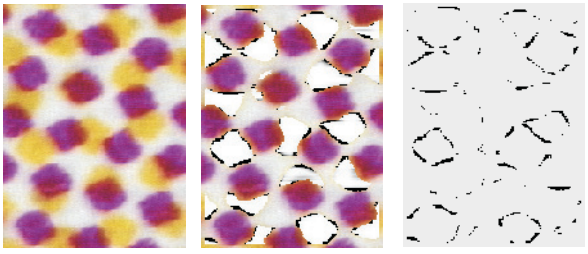


Fig.2 Edges of yellow regions

On the fig.2 see the original colour image and the images with significant edges of yellow colour between yellow-magenta and yellow-white regions.

4. Results

Many two-color samples were processed with described method. The average of segmentation error, evaluated as the difference between area value of all color regions included in the sample and 100 percent, was approximately 6 percent.

On the fig.3 see the example of cyan-magenta sample, the image of magenta edges and the image of magenta segmentation.

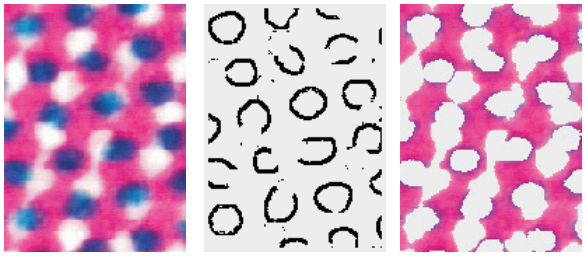


Fig.3 Segmentation of magenta regions

Improving the segmentation accuracy has the positive effect on the successful evaluation of print characteristics. The contemporary production press needs to measure, evaluate and control the print characteristics with sufficient accuracy. The method described in this work can benefit to this purpose.

References

[1] Sonka, M., Hlavac, V., Boyle, R.: Image Processing and Machine Vision. Chapmann & Hall, London 1993.

[2] Sangwine, S. J., Horne, R.E.N.: The colour Image Processing Handbook. Chapman & Hall, London 1998.

[3] Green, P.: Understanding Digital Colour. Graphic Arts Technical Foundation 1995.

[4] Russ, J. C.: The Image Processing Handbook. CRC Press, IEEE Press 1999.

[5] Fribert, M.: Separation Model of Colour Regions in a Halftone Print. Computers & Graphics 27 (2003) p.801-806.

Real-time image processing in robot soccer

Michal Gajdušek Jakub Hrabec František Šolc
Brno University of Technology, Faculty of Electrical Engineering and Communication
Department of Control and Instrumentation
Kolejní 2906/4
612 00, Brno, Czech Republic
gajdusekm@post.cz hrabec@feec.vutbr.cz solc@feec.vutbr.cz

ABSTRACT

This paper deals with design of fast image processing necessary for robot soccer system of MIROSOT category. The robots are small cubes with color patches on their top. A color camera situated over the playground captures the image of the playground. The purpose of image processing is to acquire the position and orientation of the robots in appropriately short time (the frame rate is 50 fps) from the image. In this paper we present our approach to this problem. Our techniques of image segmentation, objects classification and scene understanding are optimized for speed and robustness. We also present various types of the patches and methods for their recognition.

Keywords

Real-time, image processing, robot-soccer, MIROSOT, camera

1. INTRODUCTION

MIROSOT is one of the robot soccer categories of FIRA [FIRA04]. The robots play with the orange golf ball on the black playground (Fig. 1). Robot team patches (blue and yellow, of the minimal size 3.5 cm x 3.5 cm) are used for vision system to determine which robot belongs to which team. Possible additional color patches have to be of different color from ball and opposite team. The number of robots depends on the played league and varies from 5 to 11 robots of one team. A camera over the playground with a digitizer is a feedback of this system. We use an analog camera (with resolution 760 x 285 pixels) and a frame grabber with A/D conversion. The image from this system has more noise and distortion compared to a digital camera. Hence our approach is also robustness oriented. The robots can move up to 4 $m.s^{-1}$, therefore, the control loop should have a high-speed sampling rate with as minimal transport delay as possible. Sampling rate (50 fps) is, in our case, produced by used camera. Transport delay (35 ms) is mainly caused by mentioned feedback.

Conference proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

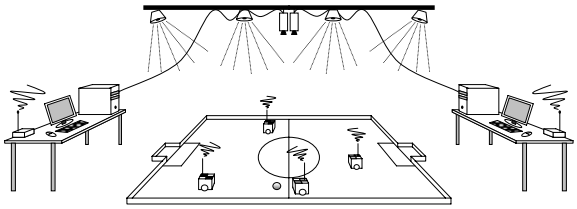


Figure 1. Overall system

2. IMAGE SEGMENTATION

We chose the color segmentation to reduce amount of data as the best way of image segmentation, because the color patches on the robots are used. We use 24-bit RGB model because of its hardware support and easy way of working with the color components. In the RGB color space every color patch is pictured as an object with irregular shape. To classify these objects in the RGB color space, we created six ranges for each class. Three ranges limit R, G and B values. Another three ranges limit ratios R/G, R/B and G/B. These six ranges are enough for fast and suitable classification.

3. OBJECT CLASSIFICATION

In this part we need to put together pixels of the same class, which belong to one patch and merge the patches to determine robot position. In the first step every fourth pixel is taken in horizontal and vertical

direction to speed up computation and skip most of the noise. If this pixel has assigned any class (it is not the background), algorithm will check if the sum of neighboring pixels of the same class in horizontal and vertical direction from this pixel is more than a chosen threshold. Then the object will be "filled" from the found pixel to find out its area. If the area and the horizontal and vertical sizes are in the chosen limits, the object is stored in the Table.

The merging process, in case of using standard identification recommended by FIRA, is always searching for two objects, which identify a robot. We must consider the distance between the objects and between old and possible new position of the robot.

4. TYPES OF THE PATCHES

It is only on our imagination, limited by the rules, to create a new identification of the robots. We could define three main types of identification of patches.

Differentiation using only colors

This is widely used identification of robots. The number of color patches on one robot varies from two (recommended by FIRA) to four and depends on team's preference. Merging these patches into the robot is similar to the algorithm described in section 2. We must only check more possible combinations.

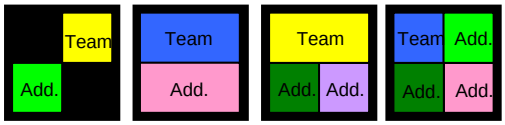


Figure 2. Types of the patches for identification by color

Differentiation using only shapes

Used shapes should fulfill several conditions: incommutability, irregularity, simple identification, noise immunity and limited minimal area. We created irregular n-angle shapes, which fulfill mentioned conditions (Fig. 3). Our shapes are identifiable by sequence of vertex angles.

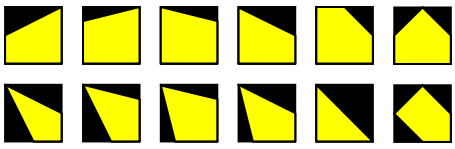


Figure 3. Possible types of the patches for identification by shape

To identify the shapes we used algorithm [Lak04a], which cyclically reduce number of the points of the shape edge. Vertex points of the shape are result of this algorithm. It is simple to get vertex angles from these points. This algorithm is suitably fast for our application, but not always reliable.

Differentiation using shapes and colors

The third type of the patches combines features of previous two types. We may create lots of unique combinations using only a few colors and a few shapes for the patches. We designed 3 shapes: a square, rectangle and L-shape. Every patch has a complement in other patch with other shape (with other color) and together they constitute the square (Fig. 4). This ordination is very robust because of redundancy of the information.

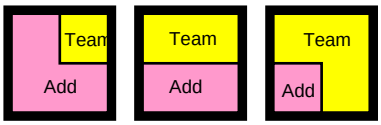


Figure 4. Identification by shapes and colors

The shapes could be differentiated by each of this parameter: area, circumference and sum of sizes in vertical and horizontal direction. The most probable shape class is determined with a nearest-neighbor method in the space of these parameters. Merging objects together is similar to the first mentioned type. Only one new condition (two objects should create a square) was added. We have very good results with this marking. Object searching is both fast and robust.

5. CONCLUSIONS AND RESULTS

This paper has presented a design of the fast image processing in the robot soccer system. We showed the methods how to locate the positions of the robots in the image. We also presented the main types of identification for the robots and outlined the methods for finding positions of the robots with these patches. Now we are using the third described type and we are able to find position of 14 robots in the image in 10 ms. The method was tested on the large playground, using a computer with P4 at 2.4 GHz and RDRAM 1066.

6. ACKNOWLEDGMENTS

This work was supported by the Ministry of Education of the Czech Republic under Project LN00B096.

7. REFERENCES

[FIRA04] FIRA. Federation of International Robosoccer Association, http://www.fira.net/ (accessed on 15th Oct. 2004).

[Pla00a] Plataniotis, K.N., Venetsanopoulos, A.N.: Color Image Processing and Applications, 2000, ISBN 3-540-66953-1

[Lak04a] Lakaemper, R., Latecki, L.J. Shape similarity project, http://knight.cis.temple.edu/ ~shape/shape/ (accessed on 10th Oct. 2004).

A Data Structure for the Construction and Navigation of 3D Voronoi and Delaunay Cell Complexes

Christopher Gold[1] christophergold@voronoi.com
Hugo Ledoux[2] hugo.ledoux@gmail.com
Marcin Dzieszko[2] marcindzieszko@wp.pl

[1] School of Computing, University of Glamorgan, Pontypridd CF37 1DL, Wales, UK
[2] Department of Land Surveying & Geo-Informatics, Hong Kong Polytechnic University, Hong Kong.

1 INTRODUCTION

The Voronoi diagram (VD) and the Delaunay triangulation (DT) can be used for modelling different kinds of data for different purposes. These two structures are attractive alternatives to rasters to discretise a continuous phenomenon such as the percentage of gold in the soil, the temperature of the water, or the elevation of a terrain. They can also be used to represent the boundaries of real-world features, for example geological modelling of strata or cadastral models of apartment buildings. The VD and the DT are an appealing solution because of their duality (they represent the same thing, just from a different point of view) and because both structures have interesting properties (see Aurenhammer [Aur91] for a review of the properties and potential applications).

Most of the algorithms and implementations available to construct the three-dimensional VD/DT store only the DT and perform their topological operations on tetrahedra, and if needed the VD is extracted afterwards. Although this results in a faster implementation, it has major drawbacks if one wants to work with the VD. It is for example impossible to assign attributes to Voronoi vertices or faces, and moreover the computation (extraction) of the VD is an expensive operation. We believe that in many real-world applications, the major constraint is not the speed of construction of the topological models of large number of points, but rather the ability to interactively construct, edit (by deleting or moving certain points) and query (interpolation, extraction of implicit surfaces, etc.) the desired model. We also think that there are many reasons that justify preserving simultaneously both the VD and the DT. The two-dimensional case has already been elegantly solved with the *quad-edge* data structures of Guibas and Stolfi [GS85]. The structure permits the storage of any primal and dual subdivisions of a two-dimensional manifold. Dobkin and Laszlo [DL89] have generalized the ideas behind the quad-edge structure to preserve the primal and dual subdivisions of a three-dimensional manifold. Their structure, the *facet-edge*, as is the case for the quad-edge, comes with an algebra to navigate through a subdivision and with primitives construction operators. Unlike the quad-edge that is being used in many implementations of the 2D VD/DT, the facet-edge has been found difficult to implement in practice and, to our knowledge, has not been used in 'real projects'. Other data structures, e.g. see [Lie94, LT97], can usually store only one subdivision.

To store and manipulate three-dimensional cells complexes, we propose a simpler structure, based on the quad-edge, that we name *augmented quad-edge* (AQE). Each cell of a complex is constructed using the usual quad-edge structure, and the cells are linked together by the dual edge that penetrates the face shared by two cells. When some basic navigation and construction operators are added to the structure, it is possible to construct and store simultaneously the 3D VD and its dual the DT. While using somewhat more storage, the approach has the advantage of conceptual simplicity and involves only a simple extension of the 2D topological relationships.

POSTERS proceedings ISBN80-903100-8-7
WSCG'2005, January 31–February 4, 2005
Plzen, Czech Republic.

2 AUGMENTED QUAD-EDGE

The quad-edge data structure [GS85] as a representation of one geometrical edge consists of four *quads* which point to two vertices of an edge and two neighbouring faces. It allows navigation from edge to edge of a connected graph embedded in a 2-manifold. Its advantages are firstly that there is no distinction be-

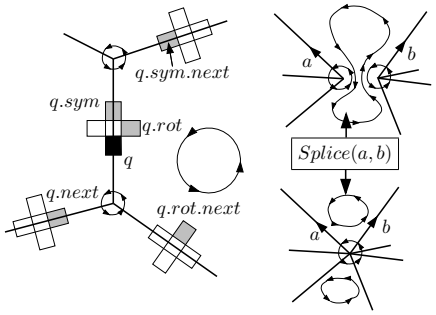


Figure 1: Left: The quad-edge structure and some basic operators. The starting quad q is the black quad, and the resulting quads are grey. Right: The *Splice* operator.

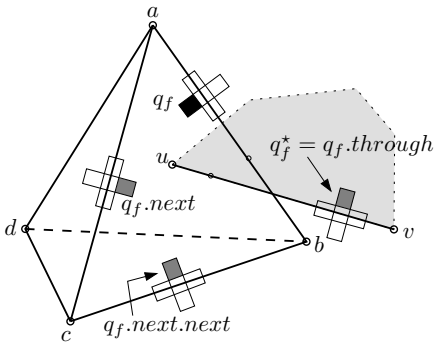


Figure 2: The *through* pointer.

tween the primal and the dual representation, and secondly that all operations are performed as pointer operations only, thus giving an algebraic representation to its operations. Figure 1 shows the basic structure, the different navigation operators (*next*,*rot* and *org*), and the construction operators *Splice* and *MakeEdge*.

The augmented quad-edge (AQE) uses the 'normal' quad-edge, which is valid for any 2-manifolds, to represent each cell of a 3D complex, in either space. For instance, each tetrahedron and each Voronoi cell are independently represented with the quad-edge, which is akin to a boundary representation (b-rep). With this simple structure, it is possible to navigate within a single cell with the quad-edge operators, but in order to do the same for a 3D complex two things are missing: a ways to 'link' adjacent cells in a given space, and also a mechanism to navigate to the dual space. First, notice that in this case two of the four *org* pointers of a quad-edge point to vertices forming the 2-manifold, but the other two (which in 2D point to the dual, or a face) are not used in 3D. Notice also that in 3D the dual of a face is an edge. Our idea is therefore to use this dual edge to 'link' two cells sharing a face: the unused face pointers simply point to their dual edge. This permits us to 'link' cells together in either space, and also to navigate from a space to its dual. Indeed, we may move from any quad-edge with a face pointer to a quad-edge in the dual cell complex, and from there we may return to a different 2-manifold in the original cell complex if needed.

A quad-edge is divided into four quads q, and there exist two types of quads: a q_f points to a face, and a q_v points to a vertex. One *rot* operation applied to a q_f returns a q_v, and vice-versa. A q_f identifies uniquely, like Dobkin and Laszlo's structure [DL89], a pair (face, edge). Therefore q_f has also a linked quad $q_f^\star$ in the dual that is defined by (edge*, face*).

One issue remains to be resolved: as each face is penetrated by several dual edges, a consistent rule must be defined to select the appropriate one. Indeed, with the AQE, the dual edge to a face has to be stored for all the dual cells sharing that edge. A triangular face has for example three dual edges since each of its three vertices becomes a cell in the dual. A q_v has its *org* pointer set to a node, and a q_f has its *org* pointer set to $q_f^\star$. The pointer to $q_f^\star$ from q_f is called *through* (Figure 2). The quad $q_f^\star$ is defined as belonging to the dual cell which encloses the node pointed to by $q_f.rot = q_v$. This is sufficient to define the *through* pointer structure.

Acknowledgements. We would like to thank the support from Hong Kong's Research Grants Council, Hong Kong (Project PolyU 5068/00E and through a research studentship to the second author).

References

[Aur91] F. Aurenhammer. Voronoi diagrams: A survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, 1991.

[DL89] D. P. Dobkin and M. J. Laszlo. Primitives for the manipulation of three-dimensional subdivisions. *Algorithmica*, 4:3–32, 1989.

[GS85] L. J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Transactions on Graphics*, 4:74–123, 1985.

[Lie94] P. Lienhardt. N-dimensional generalized combinatorial maps and cellular quasi-manifolds. *International Journal of Computational Geometry and Applications*, 4(3):275–324, 1994.

[LT97] H. Lopes and G. Tavares. Structural operators for modeling 3-manifolds. In *Proceedings 4th ACM Symposium on Solid Modeling and Applications*, pages 10–18, Atlanta, Georgia, USA, 1997.

DaMaViS – Data Management and Visualization System

Hans Hagen
Department of Computer Science
University of Kaiserslautern
Germany
hagen@informatik.uni-kl.de

Michael Münchhofen
ProCAEss GmbH
Landau
Germany
michael@muenchhofen.de

Maja Ruby
Department of Computer Science
University of Kaiserslautern
Germany
ruby@informatik.uni-kl.de

Inga Scheler
Development Acency
Rheinland-Pfalz
Kaiserslautern
Germany
scheler@ea.rlp.de

Michael Wadlé
Department of Computer Science
University of Kaiserslautern
Germany
wadle@informatik.uni-kl.de

Frank Michel
Department of Computer Science
University of Kaiserslautern
Germany
fmichel@rhrk.uni-kl.de

ABSTRACT

We describe a component based data management and visualization system called **DaMaViS**. Important aspects of our system are the component based client-server-architecture and its flexibility and reusability that are guaranteed by applying generic data structures for storing and representing data as well as using CORBA (Common Object Request Broker Architecture) for the communication between server and clients. With **DaMaViS** it is possible to handle almost every kind of data.

Keywords

Data management, visualization, virtual environments

1. INTRODUCTION

Nowadays almost every application area has to deal with a large amount of data. These masses of data need to be managed by a data management system in order to cope with complexity.

2. STATE OF THE ART

Today's data management systems are specialized to handle data of a certain application area. In the application area of environmental planning and urban development several specialized geographic information systems like land information systems, spatial information systems, environment information systems, network information systems or branch information systems do exist. Due to the large variety of systems and the huge amount of data these systems have to handle, it is necessary to use data management systems. Nevertheless, a data management system, which is able to cope with almost all application areas, is still missing.

Posters proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

3. DAMAVIS

As shown in Figure 1 the structure of **DaMaViS** is based on a client-server-architecture. We use the middleware CORBA for location and language independent communication of all components.

The client represents the user interface for communication with the server. On the left side the structure of the current project is presented as a tree to the user. The right side of the client consists of a tabbed pane, which contains the configuration and preview panel for the currently selected object.

The server can be seen as the central part of the data management system **DaMaViS**. To create maps out of GIS data an external program MapServer is integrated. The creation of a three dimensional

walkthrough is realized with a newly developed viewer. The flexibility and reusability of **DaMaViS** is increased by using JDBC for accessing the databases.

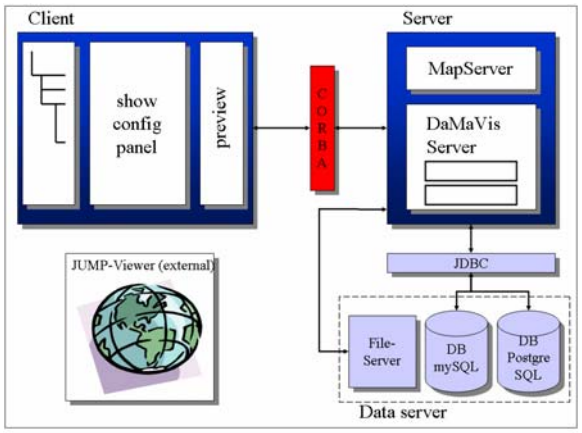


Figure 1. System structure of DaMaViS

To store GIS data and other data the database software PostgreSQL was chosen. The additional package PostGIS enables PostgreSQL to store GIS data with regard to the OpenGIS standards [Ogc04a].

4. APPLICATIONS

4.1 2D: GIS Report

Based on GIS data stored in the database a report ought to be created. Within that report a map based on GIS data shall be included. Besides maps it is also possible to add other elements like headings and text modules. **DaMaViS** provides several templates for headings and text modules.

In the end the user is able to save his constructed GIS report in several output formats. One important format for saving the created report is Microsoft Word XML format, which offers the possibility for further processing (printing, formatting …).

4.2 3D: Walkthrough

One is able to build a 3D walkthrough of interesting city quarters based on a digital terrain model and building outlines given in Gauß-Krüger coordinates. After adding a walkthrough the user has to select an appropriate digital terrain model. Then the user is able to select buildings for the walkthrough out of a city map. If a building has been added to the walkthrough, auxiliary information like height, address and textures can be chosen. The user is also able to mark the current building as a point of interest (POI). Later these points of interest are labeled with arrows in the walkthrough. For example a train station or a post office can be marked as POI.

All data of the 3D model is saved as Open Scene Graph files (OSG).

To visualize the walkthrough **DaMaViS** includes a viewer application based on the Open Scene Graph graphics toolkit. The original OSG viewer has been augmented with the ability to display all given points of interest (Figure 2). Therefore easy navigation to all interesting points of the walkthrough is provided.

Figure 2. Walkthrough with POI

The virtual walkthrough generated by **DaMaViS** could also be used for exploring world cultural heritages. One possibility would be a virtual and interactive walkthrough of Völklingen's old ironworks [Mue01a], [Mue02a].

5. FUTURE WORK

Almost every kind of data type can be integrated into **DaMaViS**. Multimedia data like videos or sounds could be associated in the GIS report and the walkthrough. For example one should be able to select a restaurant (POI) in the walkthrough and watch a short video about it.

To facilitate the creation of textures for buildings of a virtual walkthrough a tool is currently developed. This tool will interactively extract textures and create 3D models out of perspective photos taken from buildings.

6. REFERENCES

[Mue01a] M. Münchhofen; I. Scheler; K. Hergenröther and H. Hagen: Diversified Visualization from WWW to VR *International Symposium on Virtual and Augmented Architecture, Dublin* (2001)

[Mue02a] M. Münchhofen; I. Scheler; and H. Hagen: A Framework for Information Visualization; *CODATA Workshop, Paris* (2002)

[Ogc04a] Open Geospatial Consortium, http://www.opengeospatial.org

Functional Programming of Geometric Algebra and its Ray-Tracing Application

Charneau S., Aveneau L., Fuchs L.
SIC, CNRS FRE 2731, SP2MI
Bd Marie et Pierre Curie, BP 30179
86962 Futuroscope Chasseneuil Cedex – France
{charneau,aveneau,fuchs}@sic.univ-poitiers.fr

ABSTRACT

In computer graphics, geometric algebra provides a formal way to do complex geometric calculations. We propose an implementation of the geometric algebra by using a functional programming language. To emphasize its efficiency, we compare it with a well known geometric algebra implementation in an application to a ray tracer.

Keywords
geometric algebra, geometric algorithms, functional programming, imperative programming

1. INTRODUCTION

Geometric Algebra initially comes from Clifford Algebra. It has been studied to develop a mathematical system designed for a universal geometric calculus [Hes86]. The possibilities of this system are large. First, it embeds some schemes like vector algebra and quaternions in a unique system. Then it permits to represent geometric concepts by symbolic terms. Finally, specifications of operations are coordinate free, and can be easily generalised in all dimensions.

The use of the geometric algebra in computer science appears very promising, but its implementation causes some problems [PHF04], which require some cares about the data structures definition.

We present here an implementation of geometric algebra with the functional programming language Objective Caml ; we then show the results of its comparison to Gaigen [BFD03], the most efficient geometric algebra implementation that exists [PHF04].

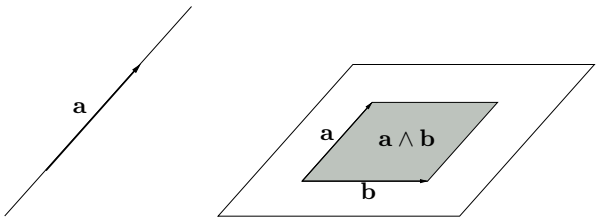
2. GEOMETRIC ALGEBRA

A geometric algebra is an associative algebra generated by a real vector space, in which a vector squares to a scalar [Hes86]. Given four vectors $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ and $\mathbf{d}$, we can define a geometric product by :

$$\begin{aligned} \mathbf{a}(\mathbf{bc}) &= (\mathbf{ab})\mathbf{c} \\ (\mathbf{a}+\mathbf{b})(\mathbf{c}+\mathbf{d}) &= \mathbf{ac}+\mathbf{ad}+\mathbf{bc}+\mathbf{bd} \\ \mathbf{a}^2 &= \epsilon_{\mathbf{a}}|\mathbf{a}|^2 \end{aligned}$$

where $|\mathbf{a}|$ is the magnitude of $\mathbf{a}$ and $\epsilon_{\mathbf{a}} \in \{-1, 0, 1\}$ is called the signature of $\mathbf{a}$.

It is convenient to decompose this product into its symmetric and antisymmetric parts by $\mathbf{ab} = \mathbf{a} \cdot \mathbf{b} + \mathbf{a} \wedge \mathbf{b}$, where $\mathbf{a} \cdot \mathbf{b}$ is a scalar and $\mathbf{a} \wedge \mathbf{b}$ is a new entity called a bivector, or 2-vector, geometrically interpreted by an oriented plane segment in the same way that a vector is an oriented line segment (see figure 1).



(a) vector, gives a unique line

(b) bivector, gives a unique plane

Figure 1: Interpretation of k-vectors

With a n dimensional euclidean space $\mathcal{E}^n$, the geometric product leads to a 2^n dimensional linear space $\mathcal{G}_n$ of multivectors constructed by a direct sum of $n+1$ linear spaces $\mathcal{G}_n^k$ of k-vectors, $k \in [0; n]$, where k is called the *grade*. 0-vectors are scalars and 1-vectors are isomorph

WSCG 2005 Posters proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

to vectors of $\mathcal{E}^n$. A k-vector $\mathbf{v}_1 \wedge \mathbf{v}_2 \cdots \wedge \mathbf{v}_k$ is geometrically interpreted by an oriented segment of the k-subspace of $\mathcal{E}^n$ generated by the k vectors $\mathbf{v}_1, \ldots \mathbf{v}_k$.

3. A FUNCTIONAL GEOMETRIC ALGEBRA IMPLEMENTATION

For our implementation, we use the functional language Objective Caml [INRIA], and we include optimizations equivalent to those found in the Gaigen implementations.
Gaigen [BFD03] is a C++ geometric algebra implementation generator. It includes some optimisations on the data structures and the operations, that consist in representing a multivector only by its non-null grade parts and in making specific treatments on them. However, doing that with an imperative language makes the data structures and their handling complex.

Interests for a functional language

The interests for a functional language are due to its links with mathematical and formal models which make the programming more efficient. Likewise, these languages were developed to handle formal expressions, which is precisely what we do in geometric algebra. Moreover Objective Caml allows *pattern matching* oriented programming, a kind of term unification system which highly facilitates term handling.

Our Implementation

We have implemented two kinds of O'Caml modules. The first one is a module that represents the geometric algebra of a given euclidean space. On this pattern, three modules were developed, for three algebras frequently used for 3 dimensional euclidean geometry. These modules integrate the same optimisations on the represented data as the Gaigen ones.
For the second implementation, we have parameterized the module by a space (the latter being specified in an other module), thanks to which every algebra can be "generated" and used, whatever the space used.

Results and comparisons

Our O'Caml implementation represent several advantages over the Gaigen libraries. First, from the point of view of the programming :

- the code is shorter and more readable,
- the data structure is simpler to define and to handle,
- we do not need intermediate data structures to control the content of a multivector, contrary to Gaigen,
- the realization of a generic module is made easier.

All these advantages simplify the development which becomes consequently faster.
Then, from the efficiency point of view, we have compared the different implementations for the calculation of rays intersections, in a home made C ray tracer.

Figure 2 shows the results we have obtained on a scene that contains about $20\ 000$ triangles, for a 640×480 pixels image size and 4 rays per pixels. The times indicated only take into account the time spent in the different geometric algebra libraries. The difference between the remained times is due to the interface between O'Caml and C. Then one can see that we are more than 2.5 times faster than the Gaigen libraries.

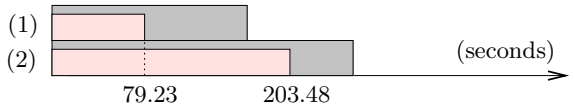


Figure 2: Rendering times for each implementation, (1) : non generic O'Caml module (2) : Gaigen optimized library

4. CONCLUSION

We have demonstrated that functional languages are more adapted to implement geometric algebra than imperative languages. Thereby, the idea of an implementation generator as Gaigen remains interesting. Our generic parameterized module is a first state for such a goal. Indeed, based on a pseudo euclidean space, a non generic optimized module should be generated simply by integrating into it the result of each product for every combinaison of k-vectors in terms of product and sum of coordinates. The determination of such results can easily be done with the generic parameterized module. Moreover, by taking more care about the data structure and its handling, and less care about the generated code (length and redundancy), we would be able to generate libraries even more optimized.
Another possible use of functional languages consists in, for a given operation on multivectors, reducing the induced term by *lazy evaluation*. For this reduction of terms, functional languages again appear particularly handy. This should minimize the calculations on coordinates as well as the errors on these floating points calculations and the time to perform them.

5. REFERENCES

[BFD03] T. Bouma D. Fontijne, L. Dorst. Gaigen. 2003, University of Amsterdam. http://www.science.uva.nl/ga/gaigen/index.html.

[Hes86] D. Hestenes. A unified language for mathematics and physics. *Clifford Algebras and their Applications in Mathematical Physics*, chapter 1, pages 1–23. Kluwer Academic Publishers, 1986.

[INRIA] Institut National de Recherche en Informatique et en Automatique. The O'Caml language. http://www.ocaml.org/.

[PHF04] C. Perwasss D. Hildenbrand, D. Fontijne and L. Dorst. Geometric algebra and its application to computer graphics. Tutorial 3, 25^{th} Annual Conference of the EACG, September 2004.

Optimal Design of Filter Banks for Texture Discrimination

M.Ali Chaudhry
Dept. of Electrical Engg
College of Signals, NUST
Rawalpindi, Pakistan
hicom71@hotmail.com

M.Noman Jafri
Dept. of Electrical Engg
College of Signals, NUST
Rawalpindi, Pakistan
mnjafri@yahoo.com

Muid Mufti
Dept. of Electrical Engg
Univ. of Engg & Tech
Taxila, Pakistan
muid@uettaxila.edu.pk

ABSTRACT

In this paper, we present an optimum design of two-band Finite Impulse Response (FIR) Quadrature Mirror Filter bank (QMF) for maximum possible discrimination between textured images. There are several applications, which may not require reconstruction of signal from its transformed coefficients e.g. texture analysis, remote sensing etc. For such applications, features are extracted at different frequency resolution scales. Hence, it is extremely important that transformed image coefficients should be distortion less, which is not possible in practice. Therefore, we present an optimal design method for maximum possible discrimination between any particular classes of textures with minimum possible coefficient error. In order to obtain the desired results, the optimization routine adjusts relative error weighting along with passband and stopband edges for the design of symmetric response FIR filter.

Keywords

Quadrature mirror filters, Texture analysis, Perfect reconstruction, Texture discrimination, Euclidean distance.

1. INTRODUCTION

The evaluation of texture features is important for several image processing applications such as biomedical imaging, industrial applications, satellite imagery etc. There is a range of feature extraction methods for texture analysis, and it has been under consideration by numerous researchers for decades. [Tuc01a]. Statistical analysis includes gray level co-occurrence, primitive length, geometrical moment analysis etc [Pra01a]. A weakness shared by all these texture analysis schemes is that the image is analyzed at one single scale. However, this limitation can be overcome by analyzing the signal at different scales with the help of filter banks. In case of filter banks, a common requirement of design objective is that the reconstructed output signal should be a delayed replica of input signal. A classical method for designing near perfect reconstruction QMF was proposed by Johnston [Joh01a]. It consists of selecting the filter coefficients such that overall transfer function becomes an all pass filter, while simultaneously minimizing the stopband energy of the transfer function.

In case of texture analysis, several features such as energy signatures, mean, variance, entropy etc are calculated on different decomposition levels. Therefore, synthesis filters can not play their part in removing distortion effect. Hence, it is extremely important that transformed coefficients should be error free [Lai01a][Wou01a].

In this paper, we propose a very simple, but effective optimization technique for maximum possible texture discrimination by keeping minimum reconstruction error. Our routine utilizes Parks McClellan algorithm for the design of symmetric FIR filter [Opp01a].

Conference proceedings ISBN 80-903100-8-7
WSCG 2005, January 31-February 4, 2005
Plzen, Czech Republic,

This paper is organized as follows: In section 2, we introduce two-band filter banks and conditions for perfect reconstruction system. Optimization problem is defined in section 3. Computational results and conclusions are given in section 4 and 5 respectively.

2. TWO CHANNEL FILTER BANK ANALYSIS

This section briefly reviews the conditions for perfect reconstruction. In a two band QMF filter banks, the reconstructed signal is given as in Eq. (1):

$$X'(z) = T(z)X(z) + S(z)X(-z) \quad (1)$$

Careful choice of synthesis filters based upon analysis filters $\{H_o(z)=H_1(-z), F_o(z)=H_1(z)$ and $F_1(z)=-H_o(-z)\}$would set aliasing term S(z) = 0 and Eq.(1) reduces to:

$$X'(z) = T(z)X(z)$$

$$T(z) = \frac{1}{2}\left(H_0(z)F_0(z) - H_1(z)F_1(z)\right)$$

$$T(z) = \frac{1}{2}\left(H_0^2(z) - H_1^2(z)\right) \quad (2)$$

$H_o(z)$ and $H_1(z)$ are the lowpass and highpass analysis filters and $T(Z)$ in Eq. (2) is called the overall transfer function of the alias free system.

3. OPTIMIZATION OF SYMMETRIC QMF FILTER-BANKS

In case of sub-band decomposition, filtering operation is followed by decimation, which is responsible for aliasing distortion. Effect of aliasing can be suppressed by attenuating stopband ripples more heavily in Parks McClellan algorithm. Choice of linear phase will reduce our search space for filter coefficients to half. Therefore, overall objective is to find N/2 filter coefficients which can minimize the expression given by Eq (3):

$$f = \int_0^p \left(\left|H_0(e^{jw})\right|^2 + \left|H_0(e^{j(p-w)})\right|^2 - 1 \right)^2 dw \quad (3)$$

Subject to maximizing the expression given in Eq. (4).

$$D(t_i, t_j) = \sum_k \left| E_k(t_i) - E_k(t_j) \right| \quad (4)$$

Where t_i represents the i_{th} texture image and E_k represents the energy associated with the filtered coefficients of an image at k_{th} decomposition level.

4. EXPERIMENTAL RESULTS

Images used in this paper were selected randomly from publicly available Meastex texture database [Mea01a]. Each texture image has dimensions of 256x256 and decomposed to three decomposition levels giving 10-dimensional energy feature vector.

Based upon the objective function defined in Eq. (3) and (4), we have developed an optimization routine which keeps the filter order fixed and adjusts the passband and stopband edges along with the relative error weighting factor. Desired objective is to search for coefficients of a FIR filter of any fixed order N, which gives concave curve for differentiability function and convex curve for energy error function. In an optimization routine, passband and stopband edges were adjusted for each incremental change in relative error weighting to obtain the desired objective function. In order to achieve the desired results, optimization routine was executed from filter order N=9 to N=20, passband error was varied from 0.5 to 6, keeping stopband error equal to unity. Finally, an optimization routine converged to the desired results for filter order N=20 as shown in Figure 1.

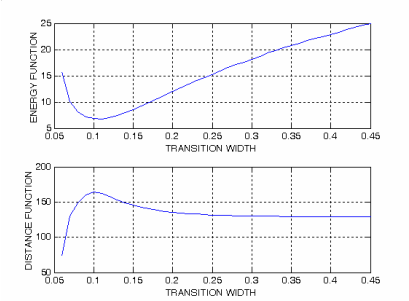


Figure 1: Energy and Differentiability Function

From Figure 1, it can be observed that the maxima and minima of the two functions coincide to a single point with respect to transition width on horizontal axis. These results are compiled in Table 1.

Filter order N	Minima of Energy Fn	Maxima of Diff. Fn	Trans. Width
20	6.79	164	$0.1p$

Table 1. Performance Comparison of Optimal Filters

Figure 2 shows the frequency response of an optimal filter (N=20

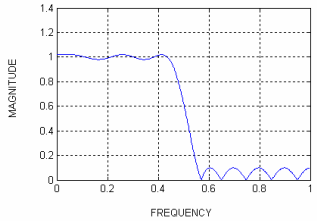


Figure 2. Frequency Response of an Optimal Filter

For performance comparison, distance between the feature vectors were calculated by using proposed optimal filter and Daubechies filter (length 20). Results are compiled in Table 2, these results show that performance of optimized filter is much better than Daubechies filter (Length 20) in terms of discriminating the texture images.

	Optimal Filter	Daubechies
Grass & Concrete	655	539
Grass & Sand	1225	951
Grass & Pebbles	636	491
Concrete & Sand	570	412
Concrete & Pebbles	21	49
Sand & Pebbles	589	460

Table 2. Comparison of Results

5. CONCLUSION AND DISCUSSION

The main concept in the analysis of signal by using filter banks is to divide the signal into its frequency contents accurately at each decomposition level. This can only be achieved with an ideal filter having a brick wall response. In practice, it is not possible to achieve such a response. Therefore, we always design a filter which performs close to an ideal case. It is observed that filter with narrow transition width performs better. Therefore, it is concluded that, symmetric impulse response filter with narrow transition width and reduced ripples in passband generally performs better as compared to orthogonal filter with greater number of vanishing moments.

6. REFERENCES

[Tuc01a] Tuceyran.M and Jain.A.K, Texture analysis, in Handbook of Pattern Recognition and Computer Vision, C.H. Chen, L.F. Pau and P.S.P. Wang (Eds.), chapter 2, 235-276, World Scientific, Singapore, 1993.

[pra01a] Pratt W. K, Digital Image Processing, John Wiley, New York, 1991.

[Joh01a] Johnston J. D., "A Filter Family Designed for QMF Filter Banks", *Proc. IEEE Int Conf Acoust Speech and Signal Proc.* Pp 291-294, April 1980.

[Lai01a] Laine A and J.Fan, "Texture Classification by Wavelet Packet Signatures", *IEEE Trans. on Pattern Anal Machine Intell,* Vol. 15, Nov 1993.

[Wou01a] Wouwer G.V, P. Scheunders and D. V. Dyck, "Statistical Texture Characterization from Discrete Wavelet Representation", *IEEE Trans. on Image Processing,* Vol. 8, No 4, April 1999.

[Opp01a] Oppenheim A.V, R.W.Schaffer and J.R.Buck, Discrete Time Signal Processing, Prentice Hall, New Jersey, 2001.

[Moj01a] Mojsilovic.A, M.V. Popovic and D. M. Rackov, "On the Selection of Optimal Wavelet Basis for Texture Characterization", *IEEE Trans. on Image Processing,* Vol. 9, No 12, December 2000.

[Mea01a] Meastex database: http://www.cssip.elec.uq.edu.au/~guy/meastex

Human Machine Interface Concept For Virtual Reality Applications

Marc Chevaldonné (1, 2, 3)
marc.chevaldonne@eads.net

Marc Neveu (1)
marc.neveu@u-bourgogne.fr

Frédéric Mérienne (2)
frederic.merienne@cluny.ensam.fr

Michel Dureigne (3)
michel.dureigne@eads.net

Nicolas Chevassus (3)
nicolas.chevassus@eads.net

François Guillaume (3)
francois.guillaume@eads.net

1 : Université de Bourgogne
UFR Sciences et Techniques
Aile de l'Ingénieur – BP 47870
21078 Dijon Cedex France

2 : Institut Image
2, rue Thomas Dumorey
71321 Chalon Sur Saône
France

3 : EADS-CCR
12, rue Pasteur, BP 76
92152 Suresnes Cedex
France

ABSTRACT

Advanced human machine interfaces, like aircraft cockpits, are difficult and expensive to design. The use of Virtual Reality technologies during the design process of the interface through early ergonomics and layout analysis can help to greatly reduce the time of development and the use of costly physical mock-ups.
Nevertheless, the use of Virtual Reality and the development of the digital mock-up must be done very carefuly.
Via the example of the Virtual Cockpit application for ergonomics studies of aircraft cockpits, this paper presents an approach which consists of a parallel study of the human-machine interface and its image: the "virtual human"-"virtual machine" interface. It introduces a conceptual model of the interface and its image, allowing the preparation of an adapted Virtual Reality application, by studying the application, the human and the machine.

Keywords

Human Machine Interface, Virtual Reality, Virtual Cockpit, Ergonomics Studies.

1. INTRODUCTION

A cockpit is an advanced human-machine interface, i.e. a complex tool of communication between a user and a machine. Its efficiency is linked to its quality, to ergonomics and to its comfort.
During the design process, a cockpit has to be tested with mock-ups. Traditionally, these mock-ups are physical: they need time to be developed and are expensive. The use of virtual mock-ups (VMU),with the help of Virtual Reality tools, instead of physical mock-ups (PMU) is greatly advantageous (easy communications between different disciplines, and saving of time and money).The development of a Virtual Cockpit application without the use of PMU, in order to simulate and provide a model and valuable information (principally for ergonomics), will help to test the interface of the cockpit very early during the design and development process. Such an application can also be used for first learning.
But the construction of the VMU and the application to test it are very complex. The use of the conceptual model of the interface presented in this paper allows a better consideration of all of these aspects, via the example of the Virtual Cockpit.
Virtual Reality tools used in this application are: a Head-Mounted Display; a glove to track the hand and fingers' movements; and a tracking system for the movement of the head, hands, and chest.

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. CONCEPTUAL MODEL

In the model presented here, the machine corresponds to the final product (e.g. the cockpit) which has to be tested, and not to the Virtual Reality tools or computers (which are only considered as tools).
We can distinguish now two flows between the human and the machine: the first one is going from

the machine to the human and is the observation flow, and the second one, going from the human to the machine is the command flow.
Two cases must be studied in parallel: the interface between the user and the machine when a PMU is used, and when the VMU is used (see Fig.1).

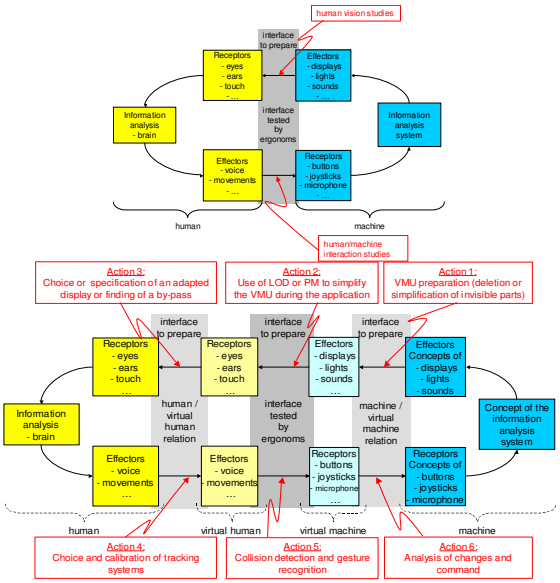


Figure 1. Human Machine interface with a PMU (upper part) or a VMU (lower part)

When using a virtual prototype, there is no physical relation between the user and the machine. The idea is then to study the relation between a "virtual human" and a "virtual machine". This relation will be the center of ergonomics studies of the human/machine interface. It is like an image of the real human/machine relation: the user is in the real world, the machine is in a "CAD/CAM world". The virtual human is the ghost of the user in a virtual world and the virtual machine is the ghost of the machine in the same virtual world. This virtual world is the bridge between two separate worlds, and is the only place where the interface can be studied.
Thus, the interface is broken up into three new ones: the human/virtual human interface (which must allow the user to command the virtual human and to receive feedback from it), the virtual machine/machine interface (which will define a simplified machine even if it doesn't appear as simplified to the user's eyes), and the virtual human/virtual machine interface (which will be used to test the human/machine interface, and then must be similar to the human/machine interface in the case of the use of a PMU).

3. DEVELOPMENT

In order to realize the Virtual Cockpit application, using this model of human machine interface, we have to study how every link between effectors and receptors of the entities should be realized.
First, the human/machine interface when a PMU is used is studied carefully. For instance, for the observation flow, we studied how and where the user looks at the machine (lines of sight), but although which are the necessary vision characteristics (field of view, visual acuity, temporal resolution...) [Che04a]. These studies will lead the realizations of the links of the interface when a VMU is used.
Hardware and software are only tools, but the application will be dependent on their capacities. That is why they have to be chosen or developed carefully. For instance, the digital mock-up is highly detailed and it is impossible to display it with so many details at a high frame rate. It is then necessary to modify it in order to avoid this issue. With the help of the studies led on the interface when a PMU is used, we delete or simplify automatically a lot of parts of the VMU [Che04a] (see Action 1 in Fig.1). During the application, Level Of Details (LOD) [Lue03a] or Progressive Meshes (PM) [Hop96a] can also be used in order to guaranty a sufficient frame rate, in function of the virtual human line of sight [Red97a] and the vision characteristics found in the studies (see Action 2). Moreover, the immerged human must be able to see the virtual cockpit as the virtual human. Therefore, an adapted display has to be chosen (see Action 3 in Fig.1), once more in function of the human vision. Our work has focused mainly on the observation flow, but will be centered on the command flow in the future (see Fig.1).

4. CONCLUSION

During the design and development process of industrial human/machine interfaces, by studying a model of the interface with a PMU, it is possible to determine criteria that will be used as guidelines during the development of the interface with a VMU, with the help of a model which is centered around the virtual human/virtual machine relation. This model is adapted to the ergonomics studies of advanced interfaces like aircraft cockpits.

5. REFERENCES

[Che04a] Chevaldonné M., Neveu M., Mérienne F., Dureigne M., Chevassus N., Guillaume F. Digital Mock-up database simplification with the help of view and application dependent criteria for industrial application. EGVE, 2004.

[Hop96a] Hoppe H. Progressive Meshes. Proceedings of ACM SIGGRAPH 1996.

[Lue03a] Luebke D., Reddy M., Cohen J.D., Varshney A., Watson B., Huebner R. Level Of Details for 3D Graphics. Morgan Kaufman 2003.

[Red97a] Reddy M. Perceptually Modulated Level Of Details for Virtual Environment. PhD Thesis (CST-134-97), University of Edinburgh 1997.

View and Application Dependent Simplification Of Digital Mock-ups

Marc Chevaldonné (1, 2, 3)
marc.chevaldonne@eads.net

Marc Neveu (1)
marc.neveu@u-bourgogne.fr

Frédéric Mérienne (2)
frederic.merienne@cluny.ensam.fr

Nicolas Chevassus (3)
nicolas.chevassus @eads.net

François Guillaume (3)
francois.guillaume@eads.net

1 : Université de Bourgogne
UFR Sciences et Techniques
Aile de l'Ingénieur – BP 47870
21078 Dijon Cedex France

2 : Institut Image
2, rue Thomas Dumorey
71321 Chalon Sur Saône
France

3 : EADS-CCR
12, rue Pasteur, BP 76
92152 Suresnes Cedex
France

ABSTRACT

The use of virtual prototypes during the industrial design process of advanced human machine interfaces, like aircraft cockpits, brings competitive advantages: the development of virtual prototypes is cheaper and faster than the development of physical prototypes; virtual prototypes support early ergonomics studies and layout analysis and can be easily enhanced all along the design and development processes.
Nevertheless, the preparation of a virtual prototype is not so easy. In industry, digital mock-ups come from CAD/CAM environment, and thus, are highly detailed. The most important challenge is then to achieve interactivity while maintaining high visual quality and fidelity.
Through the example of the Virtual Cockpit application, we present how to reduce and simplify the geometrical information of the database, in order to achieve a sufficient frame rate without degrading the geometrical visual quality. These simplifications are controlled by objective criteria based on human and application considerations.

Keywords

Virtual Reality, Virtual Cockpit, Level Of Details, Progressive Meshes, Simplification.

1. INTRODUCTION

A cockpit is a complex interface between a pilot and a flying system. Its efficiency is linked to its quality to ergonomics and to its comfort.
In order to produce an efficient interface, it is necessary to test it soon and often during the design process. That is why prototypes are needed. But the traditionally used physical mock-ups (PMU) are costly and slow to prepare. Knowing that the design of a cockpit evolves every day, another less complicated approach consists in using Virtual Reality tools: a virtual mock-up (VMU) is created from the digital mock-up (DMU), and is faster to prepare, and far much cheaper.
But VMU are not simple to prepare: for instance, in order to test the cockpit through ergonomics studies, the VMU must be able to offer the user a good visual quality and interactivity. Unfortunately, in the case of the use of Virtual Reality tools, these two characteristics are opposite: the more the VMU is detailed, the slower the application is, and vice versa. As DMU, coming from CAD/CAM design, are highly detailed, a work of preparation and adaptation of the VMU must be done in order to achieve a sufficient frame rate to guarantee interactivity. If the simplification of the DMU is based on human (visual acuity, field of view...) [CHe04a] and application (line of sight during the application) [Pai03a] criteria, it seems possible to significantly reduce the geometrical information of the DMU, to achieve interactivity and to keep sufficient visual quality simultaneously. The motive of our work is to define

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

objective criteria to guide *automatic simplifications*. The objective criteria define the limits that must not be reached in order to guarantee good spatial (linked to readability and visibility of the geometry of the VMU) and temporal (linked to interactivity and fluidity of the application) visual qualities. Automatic simplifications allow to reduce the time of preparation of the database.

We will use the example of the Virtual Cockpit application: a user is immerged in the skin of the avatar, via Virtual Reality tools (a Head-Mounted Display, a tracking system to track the movements of the head, hands and chest, and a glove tracking the movements of the fingers of the right hand). The main goal of this application is to support ergonomics studies of the interface between the future user and the future cockpit in its earliest stages.

2. SIMPLIFICATIONS

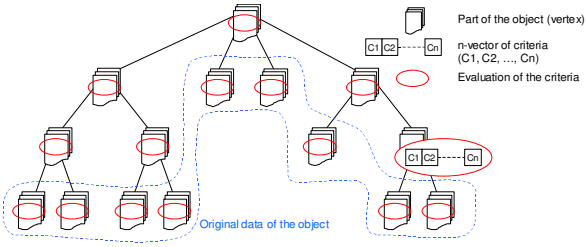


Figure 1. Use of a vertex tree structure in order to organize the objects to simplify

In order to link the geometrical characteristics and the details of the objects to the visual criteria, we structure the geometrical information in a Vertex Tree (see Fig.1) [Lue98a]. Each node of the tree represents a vertex. The leaves of the tree represent the original data of the object. If the node is a leaf, it corresponds to a vertex of the original data; otherwise it corresponds to a new vertex (into which all of its children are collapsed). For a particular state of the object, active vertices are those used in the mesh in order to represent it at this state: the higher the vertex is in the tree, the more simplified the object is. At each node of the tree, a set of criteria is tested (visual criteria, application criteria…). Our work consists in using this vertex tree introduced by Luebke [Lue98a] with our own criteria and choices of simplification [Che04a], in order to simplify the scene. We can distinguish two ways of using it: "Static simplifications" which are done before the running of the application, and "dynamic simplifications" which are performed when the application is running.

Static simplifications are preparing the geometry of the object, by keeping only the vertices that may be useful or visible by the user, by taking into account the behaviors of the user during the application. This means that a preparation must be led in order to indicate first the different possible points of view and lines of sight of the user.

Dynamic simplifications use a kind of Progressive Meshes: it changes the mesh of the object at each frame, to adapt it to the criteria. This method is well adapted to the real-time visualization because it will display details only where they are useful, and does not need any phase of preparation.

We can also use a composite simplification which consists in a static simplification, and then a dynamic simplification.

3. FIRST RESULTS

This section presents the performances of the different simplification possibilities, using a part of the Virtual Cockpit database. In order to do this, a simple OpenGL application has been programmed on a PC computer (Athlon 2200+ with 512 Mb of DDR RAM). The navigation in the database is the same in every case (61 views). The tests have been led on the original database and on simplified ones with the three methods presented (static, dynamic, composite. The results are schematically summed up in the tab below (see Tab.1).

	original cockpit	static cockpit	dynamic cockpit	composite cockpit
Preparation	No	Yes	No	Yes
Number of faces	100%	80%	≈35%	≈35%
Memory costs	100%	75%	765%	545%
Update and rendering time	100%	81%	≈49%	≈49%

Tab 1. Sum-up of the tests results of the simplification methods

4. CONCLUSION

The different methods of simplification are based on human and application criteria, which help to enhance the temporal visual quality (i.e. the frame rate), without degrading the spatial visual quality (i.e. the geometry), from the point of view of the human eyes during the application. The dynamic method seems interesting except for the memory costs. A reduction of this memory cost can be achieved by using a composite simplification.

5. REFERENCES

[Che04a] Chevaldonné M., Neveu M., Mérienne F., Dureigne M., Chevassus N., Guillaume F. Digital Mock-up database simplification with the help of view and application dependent criteria for industrial application. EGVE, 2004.

[Hop96a] Hoppe H. Progressive Meshes. Proceedings of ACM SIGGRAPH 1996.

[Lue98a] Luebke D.P. View-Dependent Simplification of Arbitrary Polygonal Environments. PhD dissertation, Chapel Hill, University of North Carolina. 1998.

Flexible Terrain Representation using Runtime Surface Reconstruction

Jan Kolar, Peer Ilsøe
Centre for 3DGI
Aalborg University
Niels Jernes Vej 14
DK-9220, Aalborg, Denmark
{kolda,ilsoe}@3dgi.dk

ABSTRACT

Terrain has a great potential as a reference for visual navigation, which can be utilized to access and manage information. From this standpoint the geometry of the terrain is a unique defining surface for all geographic applications as well as for any geo-related information. However, a data representation of terrain in three-dimensions provides numerous challenges for visualization as well as for analytical purposes. Solutions that satisfy visualization criterions often appear to be less optimal for maintenance or scalability required by analytical applications and vice versa. This work proposes a geometric data representation of the terrain that respects both visualization and analytical applications. The solution can be used for an entire planet, which allows avoiding needs for performing conversions between cartographic projections and transformations between geodetic datums. The data representation provides good local geometric flexibility like TIN but also supports multiple levels of detail. Amounts of data can grow large gradually---it is possible to alter local areas only while leaving distant parts of the terrain unchanged, which is convenient for maintenance. Introduced approach is based on runtime construction of triangulated irregular network using Delaunay triangulation. The mass points used for the surface reconstruction are structured in order to support multiple levels-of-detail. The proposed representation allows managing terrain data in numerous detached repositories, which can be used for distributed solution.

Keywords

Terrain representation, Runtime surface reconstruction, Delaunay triangulation, Global terrain, TIN, LOD.

1. INTRODUCTION

Visualization of terrain datasets with LOD capabilities is an active area of research. Two main kinds of approaches can be identified within these works. The first are based on construction of bin-tree of similar triangles that represent the surface. The second kind of approaches is based on irregular triangulated networks (TIN).

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

Since all these works assume terrain data as an offset from plane, global solution addresses additional problem. Existing solutions utilize projections of the sphere (or its part) onto a plane. Methods with projection onto one [Aas02] four [Cig03] and six [Pen04] planes have been introduced in order to provide global terrain solutions.

2. METHOD

Spatial Division Scheme

For purpose of indexing terrain globally a global grid system called geoindex has been developed [Kol04]. This approach takes advantage of the height field property of the terrain, meaning the terrain is a displacement from a simpler mathematically defined surface, such as the sphere. Geoindex tessellates the sphere into a set of cells of similar size forming a

Voronoi diagram on the sphere [Luk87]. Using tessellations with different number of cells around the sphere, geoindex can provide multiple levels of tessellation. Advantages of geoindex with respect to indexing of global terrain data are avoidance of projections onto a plane and use of a single uniform division scheme. In contrast spatial indices based on plane cannot avoid projections and usually employ a set of projected planes.

LOD Construction

A decimation technique similar to greedy insertion algorithm introduced by [Gar95] has been used for construction of discrete LOD. This approach involves an iterative process of inserting DTM points of the highest importance to a triangulation that provides an approximation of the surface. The importance of DTM point is defined as a vertical offset of the point from the decimated version of the surface at particular iteration (i.e. the importance changes over iterations). On each pass one or more DTM points are inserted to the triangulation. Insertions of DTM points are irrevocable; meaning that the resulting list of mass points is sorted according to their importance to the original DTM. This property of the result facilitates the construction of LOD.

Terrain Data Structure

Mass points are kept in a simple data structure. There is one record per mass point, which can be modeled as a six-tuple *(l, gl, gc, x, y, z)*; where *l* denotes LOD, *gl* is the geoindex level (density) used for tessellation of the sphere, *gc* stands for the code of the tessellation unit [Kol04] at the corresponding level *gl* and *x, y, z* are the Cartesian coordinates of the mass point. Data are structured according to the spatial tessellation associated with particular LOD, i.e., according to *l* and *g* properties of the six-tuple.

An extent of a spatial unit in which LOD is stored also reflects the range that is feasible for the particular LOD. This means that data from different LOD are stored in spatial units of different size, e.g., mass points that constitute lower LOD are in larger tiles than tiles used for more detailed LOD. This is a valid reasoning mainly for visualization applications in which near geometries are required to be in higher detail while still be visualized in a coarser but broader context of the surrounding terrain.

This data structure allows reuse of mass points meaning that the data from all coarser LOD are required in order to obtain complete geometry at given location.

Runtime Surface Reconstruction

At any point when navigating through the scene, the appropriate LOD is used for visualization of the terrain geometry. An effective approach to achieve this behavior is to use pieces of the scene in different LOD and combine them into a seamless surface. This is a traditional major problem for terrain geometry because there is no topological correlation between the data from different LOD. Using proposed data structure in combination with runtime surface reconstruction, however, avoids this "stitching" problem, because there is no topology stored for mass points.

Delaunay triangulation has been proven to be the optimal triangulation in 2D, fast enough to reconstruct surface from several thousands of points at runtime. However, because the global terrain solution is represented as an offset from the sphere, the plane against which the triangulation is performed changes with the viewpoint. As a solution a tangent plane at the centre of the nearest tessellation unit from the coarsest LOD is used.

3. ACKNOWLEGMENTS

This work has been done as a part of Grifinor project at Centre for 3D GeoInformation (www.3dgi.dk) at Aalborg University.

4. REFERENCES

[Aas02] Aasgaard R., Projecting a regular grid onto a sphere or ellipsoid. In Advances in Spatial Data Handling, Richardson, D. and Oosterom, van P., Springer-Verlag, pp. 339-350, 2002.

[Cig03] Cignoni P., Ganovelli F., Gobbetti E., Marton F., Ponchio F., and Scopigno R. Planet-Sized Batched Dynamic Adaptive Meshes (P-BDAM). In Proceedings IEEE Visualization. pp. 147-155. IEEE Computer Society Press, 2003.

[Gar95] Garland M., Heckbert P. Fast Polygonal Approximation of Terrain and Height Fields. Technical Report CMU-CS-95-181, 1995.

[Kol04] Kolar J., Global indexing of 3d vector geographic features. In Proceedings of International Society for Photogrammetry and Remote Sensing 20th Congres, 4, pp. 669-672, 2004.

[Luk87] Lukatela H., Hipparchus geopositioning model: an overview. In Proceedings of Eighth International Symposium on Automated Cartography (Auto-Carto 8), 1987.

[Pen04] Peng W., Petrovic D., Crawford C., Handling large terrain data in GIS. In Proceedings of International Society for Photogrammetry and Remote Sensing 20th Congres, 2004

3D Visibility of 4D Convex Polyhedra

Alexej Kolcun
Institute of Geonics Ac.Sci CR,
Studentska 1768,
708 00, Ostrava, Czech Republic
kolcun@ugn.cas.cz

ABSTRACT

In mathematics we can easily generalize Euclidean 3D space to n-dimensional one for arbitrary n>3. The task, how one can express n-dimensional objects in 3D or even in 2D, arises. In the paper the generalization of the back-volume culling algorithm is analyzed.

Keywords

visibility, higher-dimensional polyhedra

1. INTRODUCTION

The output information from the physically based models is very often in the form of spatial data set with an internal structure (e.g. vector field, tensor field). It means that the set of the values in defined point is transformed according to defined rules when the point is moved to the different position. In the paper we consider the simplest case of such structure – only *n*-dimensional Euclidean space with rotation.

In mathematics we can easily generalize Euclidean 3D space to n-dimensional one for arbitrary *n*>3. The task, how one can express *n*-dimensional objects in 3D or even in 2D, arises. In [Agu04] the unraveling approach is used, in [Hol91] Depth-Cueing of 4d bodies is applied.

In the paper the generalization of the visibility criterion for convex polyhedra is analyzed. Criterion is formulated for 4D case.

2. THE FIRST APPROACH TO THE VISIBILITY

Visualization of *n*-dimensional convex polyhedra we can realize it in two steps: **1.** projection of the body vertices to 3D, **2.** construction of the convex hull of 3D projections. As we consider convex polyhedra only, the scheme is correct. This schema is used e.g. in [Agu04], [Holl91]. However, this approach doesn't use any information about the structure of the body (faces) and its projections.

Our solution is based on the well-known criterion – back-volume culling algorithm (BVCA).

Conference proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

3. ORIENTATION AND VISIBILITY

When we use BVCA, we must distinguish between external and internal side of the faces. So, the orientation of the space must be introduced. In 3D case the ordered triplet of base vectors $(\vec{e}_1, \vec{e}_2, \vec{e}_3)$ is right oriented, if

$$[\vec{e}_1, \vec{e}_2] = \vec{e}_3, \quad [\vec{e}_2, \vec{e}_3] = \vec{e}_1, \quad [\vec{e}_3, \vec{e}_1] = \vec{e}_2 \qquad (1)$$

In 4D case we can introduce the vector product in similar way as in 3D case:

$$[\vec{u}, \vec{v}, \vec{w}] = \begin{vmatrix} \vec{e}_1 & \vec{e}_2 & \vec{e}_3 & \vec{e}_4 \\ u_1 & u_2 & u_3 & u_4 \\ v_1 & v_2 & v_3 & v_4 \\ w_1 & w_2 & w_3 & w_4 \end{vmatrix}$$

Here the ordered quadruplet of base vectors $(\vec{e}_1, \vec{e}_2, \vec{e}_3, \vec{e}_4)$ is right-oriented if the relations below are fulfilled:

$$\begin{aligned} &[\vec{e}_1, \vec{e}_3, \vec{e}_2] = \vec{e}_4, \quad [\vec{e}_1, \vec{e}_2, \vec{e}_4] = \vec{e}_3, \\ &[\vec{e}_1, \vec{e}_4, \vec{e}_3] = \vec{e}_2, \quad [\vec{e}_2, \vec{e}_3, \vec{e}_4] = \vec{e}_1. \end{aligned} \qquad (2)$$

Here $\vec{e}_i = (e_{i1}, e_{i2}, \ldots e_{in}),\ e_{ii} = 1,\ e_{ij} = 0,\ i \neq j,.$

Following example demonstrates the consistent orientation of volumes:

Let the 4D simplex is defined on the vertices 0=(0,0,0,0), 1=(1,0,0,0), 2=(0,1,0,0), 3=(0,0,1,0), 4=(0,0,0,1). Orientation of the triplets of the base vectors 1-0, 2-0, 3-0, 4-0 in (2) is illustrated as oriented triangles in Fig.1. For oriented tetrahedron sim(a,b,c,d) defined on triplet of base vectors b-a,c-a, d-a, the complementary base vector e-a defines 'the external normal vector' (arrows in Fig.1).

According to (2) we obtain right-oriented 3D sub-simplexes sim(0,1,3,2), sim(0,1,2,4), sim(0,1,3,4), sim(0,2,3,4) with external normal vectors 4-0, 3-0, 2-0, 1-0.

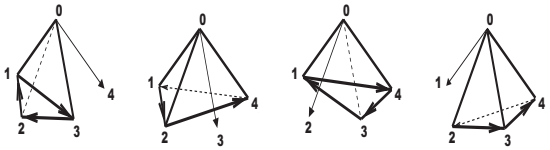


Figure 1. Right-oriented sub-simplexes.

4. VISIBILITY OF 4D BODIES

Let's consider 4D cube. Its 3D projections

$$C_{i,0} = \left\{ (x_1, x_2, x_3, x_4) \in \langle 0,1 \rangle^4 : x_i = 0 \right\},$$
$$C_{i,1} = \left\{ (x_1, x_2, x_3, x_4) \in \langle 0,1 \rangle^4 : x_i = 1 \right\}.$$

are in the Fig.2. Orientation of these 3D projections is the same as in the Fig.1. E.g. we can see that $C_{4,0}$ is in the subspace $\vec{e}_1, \vec{e}_2, \vec{e}_3$ so the sim(0,1,3,2) defines it's orientation.

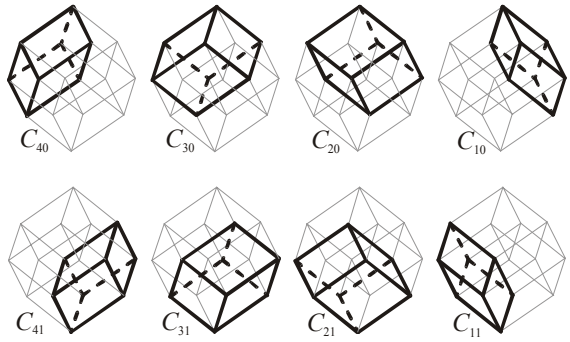


Figure2. Oriented 3D projections of 4D cube.

Let us choose the view vector so that $C_{4,0}$, $C_{2,0}$ are visible and $C_{3,0}$, $C_{1,0}$ are invisible. It can be proved that the orientation of $C_{i,0}$,$C_{i,1}$ are mutually opposite.

BVCA criterion connects the visibility of the body with the visibility of its *(n*-1*)*D projections. But we cannot see the whole 3D visible volumes − we can see only some of their faces. Moreover, there exists a 3D-visible volume of 4D-cube, which doesn't take place in resultant 2D-visible projection. So, we cannot use directly BVCA criterion when 4D-body is visualized.

We shall use next important construction – contour.

2D contour of 3D body consists of edges, which are the intersections of visible and invisible faces.

In similar way we can introduce the 3D-contour of 4D-body as a set of faces, which are the intersections of 3D-visible and 3D-invisible projections. The resultant criterion can be formulated:

face is visible <=> face is intersection of visible and invisible 3D-projections of body.

In the most general case the 3D-contour of the 4D-cube obtains 12 contour faces ($C_{i,0} \cap C_{i,1} = \emptyset$). So, the most general resultant projection of the 4D-cube is dodecahedron – see Fig. 3.

Figure 3. The most general resultant 3D representation of 4D cube.

It similar way we can formulate the visibility criterion in *n*-dimensional case:

2D-face is visible <=> face is intersection of visible and invisible (n-1)-D projections of body.

5. CONCLUSIONS

Representation of 4D cube in "usual habit" reduces original body very significantly, in similar way as the substitution of the 3D cube with its hexagonal contour.

Visibility criterion, which connects visibility of boundary faces of the body with the visibility of its n-1 dimensional projection, is introduced.

6. ACKNOWLEDGMENTS

The work is supported by project K1019101 (Czech Ac.Sci.) – Mathematics, informatics and cybernetics: the tools and the applications.

7. REFERENCES

[Agu04] Aguilera, A., Perez-Aguila, R. General n-dimensional rotations. Journal of WSCG 2004.

[Holl91] Hollasch, S. R. Four-space visualization of 4D objects. Master's thesis, Arizona State University 1991.

A scripting tool for real-time effect programming

Calle Lejdfors, Lennart Ohlsson
Department of Computer Science
University of Lund
Box 118, SE-221 00 Lund, SWEDEN
{calle.lejdfors, lennart.ohlsson}@cs.lth.se

ABSTRACT

Writing real-time visual effects for graphics hardware is made difficult by the high degree of dependence between GPU-level shaders and CPU-level orchestration of pipeline settings parameter bindings. This paper presents PyFX, an effect framework embedded in the programming language Python. Compared to existing existing frameworks this language embedding gives the effect programmer greater expressive power. These benefits, together with some imrpoved functional features of the framework, are demonstrated through some illustrative examples.

Keywords
Effect programming, CPU/GPU interaction, embedded languages

1 INTRODUCTION

With the introduction of programmable real-time hardware, procedural techniques previously used for cinematic effects have been made available for use in real-time graphics as well. However, writing real-time effects is made difficult because the CPU and GPU code is typically written in separate languages, while still having strong inter-dependencies.

One solution to this problem are so called *effect frameworks* which enable the unification of shader programs with the necessary pipeline states required for correct operation. Current effects frameworks such as DirectX Effects (DXFX) by Microsoft [2] and CgFX by NVIDIA [1] provide a number of features which simplify programming real-time visual effects. Both frameworks rely on effect specifications consisting of parameters, shaders, techniques, and passes, which are stored in text files and loaded at run-time.

However, the format used in current effect frameworks is lacking all but the most basic forms of abstractions, data-hiding, or sharing, making effect development unnecessarily complicated. Also there is lack of integrated support for some functional features such as RTT (render-to-texture) commonly used in todays graphics applications.

Posters proceedings ISBN 80-903100-8-7
WSCG'2005, January 31–February 4, 2005
Plzen, Czech Republic

2 THE PYFX FRAMEWORK

Our effect framework, PyFX, implements a complete effect programming language embedded in the object-oriented scripting language Python. The fact that PyFX is embedded in a fully fledged programming language enables high-level language features to be used for expressing effects. By using constructs such as function definitions, loops, conditionals, and modules to express and share common parts, an effect description becomes shorter and more clear. And the choice of a very high level language like Python makes it possible to achieve these benefits and still retain the declarative style of existing frameworks.

The functionality of PyFX includes all the features found in the DXFX and CgFX frameworks. In addition it provides:

- *Render-to-texture* – Render to an off-screen area which can be used as a texture.
- *Image processing support* – GPU based image processing operations can be applied to any texture or off-screen area.
- *Support for shader interfaces* – PyFX enables easy use of Cg's interfaces allowing run-time construction and composition of shader programs.

To accommodate these functional additions the PyFX language extends the passes of other effect frameworks to:

- *RenderGeometry* – the usual passes of other frameworks. Sets up the appropriate states and then instructs the application to submit geometry
- *ProcessImage* – performs 2D image processing between any number of images (which may reside in textures, off-screen areas or the current screen buffer). It provides support floating point targets and sources enabling HDR image processing.

PyFX is written in Python and built on top of OpenGL and Cg. The implementation in Python has made it possible to build a very flexible interface towards the application using the framework. This is described in more detail in [4].

3 EXAMPLES

As an example of using PyFX we present a glow effect [3], used to simulate the nimbus due to atmospheric scattering which appear around brightly lit surfaces. This effect is implemented by rendering an object to the screen, rendering the glowing parts of the object to an off-screen buffer, blurring the off-screen buffer and then additively blending the result to the screen. This can be expressed in PyFX by:

```
def RenderGlowRegions(target):
    return RenderGeometry(
              Target=target,
              VertexShader=glowMask.vs(),
              FragmentShader=glowMask.fs())

def GaussianBlur(source):
    ...

def AdditiveBlend(source, target):
    return ProcessImage(Source=source,
                        Target=target,
                        SrcBlend = SRCALPHA,
                        DestBlend = ONE)
```

The technique which performs blurring can now be written simply as

```
technique = [RenderGeometry(),
             RenderGlowRegions(blurBuffer),
             GaussianBlur(blurBuffer),
             AdditiveBlend(blurBuffer, Screen)]
```

By making use of PyFX's language and functional features the resulting effect is a readable specification of what the effect does and how it does it.

Examples of effect is simplified by introducing high-level languages features are numerous. For instance, fur typically makes use of multiple layers of decreasing opacity which are additively blended over the solid object [5]. This can easily be expressed in PyFX by using a function definitions to express the drawing of a single fur shell:

```
def RenderFurShell(s):
    shell = s/FurThickness
    return RenderGeometry(
                AlphaBlendEnable = True,
                SrcBlend = SRCALPHA,
                DestBlend = ONE,
                VertexShader = furVS(Shell=shell),
                FragmentShader = furFS(Shell=shell))
```

The entire effect, drawing the solid object followed by a number of shells, can then be expressed as

```
technique = [RenderGeometry()] + \
            [RenderFurShell(i)
                for i in range(1,NumberOfShells)]
```

again providing a clearly legible description of the operation of the effect.

4 CONCLUSIONS

We have presented an effect framework which improves on current frameworks in two respects; it enables to use of high-level language features for effects descriptions while retaining the declarative style of current effect frameworks, and, it provides an extended set of integrated functional features making a larger set of effect possible.

The main purpose of PyFX is to be a tool for investigating which features and characteristics are useful and desirable for effect programming. As such it provided a flexible environment for experimenting with effect frameworks and effect programming.

References

[1] CgFX 1.2 Overview. `http://developer.nvidia.com/`.

[2] DirectX SDK Documentation. `http://msdn.microsoft.com/`.

[3] Greg James and John O'Rorke. *GPU Gems*, chapter Real-Time Glow, page 816. Addison Wesley Professional, 1 edition, March 2004.

[4] Calle Lejdfors and Lennart Ohlsson. Pyfx - an active effect framework. In Stefan Seipel, editor, *SIGRAD 2004*, number 13 in Linköping Electronic Conference Proceedings, 2004.

[5] Jerome Lengyel, Emil Praun, Adam Finkelstein, and Hugues Hoppe. Real-time fur over arbitrary surfaces. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 227–232. ACM Press, 2001.

A comparison between contour and histogram-based observation models for tracking[1]

M.J. Lucena
Departamento de Informática
Escuela Politécnica Superior
Universidad de Jaén
mlucena@ujaen.es

J.M. Fuertes
Departamento de Informática
Escuela Politécnica Superior
Universidad de Jaén
jmf@ujaen.es

N. Pérez de la Blanca
Departamento de Ciencias de la Computación e I.A.
ETSII. Universidad de Granada
nicolas@ugr.es

ABSTRACT

In probabilistic tracking tasks, the quality of the observation model used is of prime importance. In some models, evidence is extracted from the outline of the object while in others, it is extracted from the region bounded by this. In this article, we shall study and compare the behavior of two outline-based models with one histogram-based model.

Keywords

Probabilistic tracking, dynamical model, observation model.

1.This work has been financed by grant TIC-2001-3316 from the Spanish Ministry of Science and Technology.

1. INTRODUCTION

The use of probabilistic models applied to tracking enable us to estimate the *a posteriori* probability distribution, $p(X \mid Z)$, of the set of valid configurations for the object to be tracked, represented by a vector X, from the set of measurements Z taken from the images of the sequence. The likelihood at a given time t_{k-1} is combined with a *dynamical model* giving rise to the *a priori* distribution in the next instant t_k, $p(X)$. The relation between these distributions is given by Bayes' Theorem. In order to estimate $p(Z \mid X)$, know as the *observation probability*, we will use several contour-based *observation models*, define in [Bla98] [Luc03a] [Luc03b], and one region histogram-based observation model.

POSTERS proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. OBSERVATION MODELS

Observation model based on intensity restricticions

Let $x = f(X_{t_k}; m)$ (where X_{t_k} defines the specific configuration of the object model, and m is the parameter vector which associates each point within the model with a point on the image plane), a point belonging to the model outline at the instant t_k. Let S be a neighborhood of x subdivided into S_i and S_e (corresponding respectively to the parts of the neighborhood which remain towards the inside and outside of the object outline), then we will calculate the expression:

$$d(c_{t_k}, m) = f(X_{t_k}; m) - f(X_{t_{k-1}}; m)$$

We then consider the optical flow constant in S_i and S_e, respectively, and use the system of equations proposed in [Luc81] to obtain $f_{S_x} = (f_x, f_y)$, where S_x shall be S_i or S_e, respectively. The temporal derivatives of the image are computed as

$$I_t(x) = I^{(k)}(x + d(c_{t_k}, m)) - I^{(k-1)}(x)$$

Two different flow estimations are obtained, $f_{S_i}(c_{t_k}, m)$ and $f_{S_e}(c_{t_k}, m)$, corresponding to the inner and outer area of the neighborhood of x, respectively.

The quadratic differences with the expected flow (which in this case equals zero) are equivalent to the squared norm of the estimated flow vectors:

$$Z_{S_i}(c_{t_k}, m) = \left\| f_{S_i}(c_{t_k}, m) \right\|^2$$
$$Z_{S_e}(c_{t_k}, m) = \left\| f_{S_e}(c_{t_k}, m) \right\|^2$$

These values may be combined and a value of $Z(c_{t_k}, m)$ may therefore be obtained:

$$Z(c_{t_k}, m) = \frac{Z_{S_e}(c_{t_k}, m)}{Z_{S_e}(c_{t_k}, m) + Z_{S_i}(c_{t_k}, m)}$$

We will consider that the presence probability of the measurements obtained for the image, since they have been caused by the point of the outline corresponding to the vector **m** of the sample in question, defined by the vector c_{t_k}, must be proportional to the function $Z(c_{t_k}, m)$ computed previously, and that given the independence between the different points of the outline,

$$p(Z \mid c_{t_k}, m_i) \propto Z(c_{t_k}, m_i)$$

with m_i being the vector which identifies the i-nth point on the outline of the model.

Observation model based on histogram

Given the region occupied by the outline defined by $X_{i\,i}$, we can calculate its histogram and define the observation probability according to the distance between this histogram and that of a reference model extracted from the first frame of the sequence. For this work, we have used one function for the observation probability:

$$p(Z \mid X_{t_k}) \propto \exp^{-\frac{d_H(H_r, H_{x_i})^2}{K}}$$

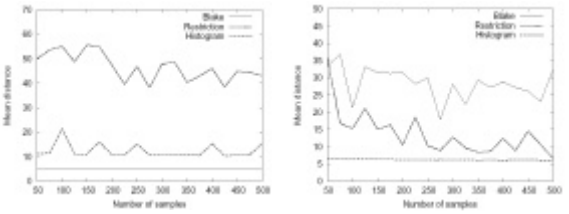
with H_r being the reference histogram, H_{x_i} the histogram corresponding to the sample, K a constant, and

$$d_H(H_r, H_{x_i}) = \sum_j \frac{(H_r(j) - H_{x_i}(j))^2}{H_r(j) + H_{x_i}(j)}$$

3. EXPERIMENTS

We have used the CONDENSATION algorithm [Bla98] on two image sequences, each lasting 10 seconds, at 25 frames per second, 320x240 pixels, and 8 bits per band and pixel. These sequences correspond to the movement of a hand over a background with and without clutter. For measuring the performance of each observation model, we have used the following procedure: given the initial frame of each sequence, we have used the Harris operator to obtain a number of points, and have manually selected five points placed on the contour of the object of interest; we have then located the corresponding points for the entire sequence. Once the tracking process has finished, we compute the mean Euclidean distance between each point and the position estimated by the tracker for that point, throughout the entire sequence.

The results obtained for the first sequence (Figure 1.a) show that Blake's model is clearly superior. Nevertheless, the scene shown in this sequence is unrealistic as it has a constant background. In the case of the second sequence (Figure 1.b), the results are very different. In the presence of clutter, Blake's model behaves much worse. The intensity restriction-based observation model performs slightly better for the second sequence than for the first.



a) b)

Figure 1. Precision obtained according to the number of samples by the different observation models for the first (a) and second (b) sequence.

4. CONCLUSIONS

In this moment, the last experiments suggest that probably, the combination of different sources of evidence applied to probabilistic tracking tasks can produce better results than models applied independently.

5. REFERENCES

[Bla98] Blake, A. and Isard, M. Active contours, Springer, 1998

[Luc81] Lucas, B. and Kanade, T. An interactive image registration technique with an application to stereo vision. In Proceedings of DARPA IU Workshop, pp. 121-130, 1981

[Luc03a] Lucena, M., Fuertes, J.M., Perez de la Blanca and Garrido, A. Using optical flow as evidence for probabilistic tracking, Lecture Notes on Computer Science, no. 2749, pp. 1045-1049, 2003

[Luc03b] Lucena, M., Fuertes, J.M., Perez de la Blanca, Garrido, A. and Gomez, J.I. Optical flow-based probabilistic tracking. In proceedings of VII ISSPA, vol. 2, pp. 219-222, 2003

Low Cost Avatars Animation System from Real Images Compliant MPEG4

B. Mir, A. Salas, A. Jaume.
Unitat de Gràfics i Visió per Ordinador – Dept. Matemàtiques i Informàtica - UIB
C/Valldemossa Km. 7.5, 07122 - Palma de Mallorca - España
Phone: (34) 971 17 27 11
antoni.jaume@uib.es

M. Bez
Departamento de Informática. - Centro Universitário Feevale
RS 239, 2755 - Cep 93352-000 - Novo Hamburgo - RS – Brasil
Phone: (55) 51586-8800
martabez@feevale.br

F. Perales.
Unitat de Gràfics i Visió per Ordinador – Dept. Matemàtiques i Informàtica - UIB
C/Valldemossa Km. 7.5, 07122 - Palma de Mallorca - España
Phone: (34) 971 17 27 11
paco.perales@uib.es

ABSTRACT

In this paper we present a low cost system to capture, modelization and animation of avatar face with a few control points based in FAP´s MPEG4 coordinates. A Facial Animation Engine is designed as high-level interface for the synthesis and animation of virtual faces that is in full compliance with MPEG-4. We implement the MPEG-4 standard specifications for the adaptation and animation of 3D wire-frames models with respect to the reproduction of two characteristics: realism in adapting the model geometry to the characteristics of any specific face and realism in performing facial expressions. Starting from a facial wire-frame model and from a set of configuration files, the developed system is capable of automatically generate the animation stream of FAPS (Facial Animation Parameters). Our system uses an Interpolation approach to emulate the behavior of face tissues. The final objective is to use the system in real time environments and for portable PDA, mobile phone applications or virtual reality applications.

Keywords

Calibration, FAP´s, MPEG4, Facial Animation Engine, Real time Applications, Interpolation.

1. INTRODUCTION

The realistic capturing, modeling and animation of the human face is one of the most elusive goals in computer vision and animation. We focus on the delivery of real time (and hence low bit rate) facial animation for use in collaborative virtual environments.

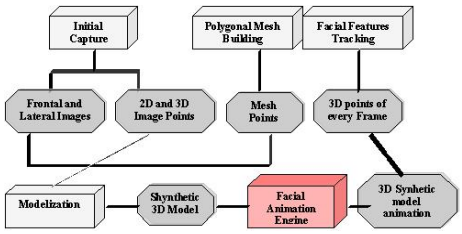


Figure 1. Whole facial system

Our objective is to achieve the best results possible using relatively low cost, widely available equipment. No Facial markers are providing the control points for facial analysis and later model deformation. At the moment the system select some points in a automatic way (lips corners, eyebrows, etc..) and other in a manual initialization procedure. Also the focus is on recognizing lip movement for vocal communication and major expressions such as smiling, frowning, surprise etc. Figure 1 explains the whole system.

2. MPEG-4 FAP's AND MODELITZATION MODULE

The MPEG-4 Facial Animation standard specifies a face model in its neutral state. 84 Feature Points (FPs), used to provide spatial reference to specific positions on a human face and 68 Facial Animation Parameters (FAPs), that move the FPs producing the animation. Feature Points are arranged in groups like cheeks, eyes, head, etc.

In this module, a *realistic* face model is constructed from a frontal and a lateral photography of a person.

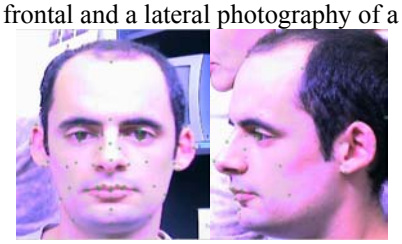


Figure 2. Frontal and a lateral picture of person

We take two pictures with two cameras that are calibrated using the Tsai calibration method. Once we have the two pictures, we can reconstruct the 3D FDP points from one side of the face. The user selects (we are working in a automatic version of key points detection using computer vision techniques) the FDP points of one side of the face. After that, the 3D

coordinates of the FDP points of the right side of the face are reconstructed using the calibration of the two cameras. Then, the FDP points of the left side of the face can be reconstructed using symmetry. We have defined a very simple symmetry plane, with the points 11.1 (forehead), 2.1 (chin) and 9.3 (nose). We get 75 of the 84 FDP points of the face. The points are joined in triangles. The result can be seen in the wireframe model. Also we can apply a texture onto the wireframe model.

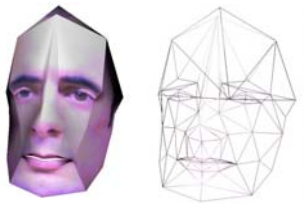


Figure 3. Result model.

2.ANIMATION MODULE

The Facial Animation Engine is designed to provide a high-level interface for building MPEG-4 compliant animated models. This interface is composed with the cores mechanisms responsible of the facial animation and is capable of animating a facial wire-frame starting from its geometric description and from the associated semantic information. The geometric description of the model to be animated is a fully compliant VRML file. In this way, we only heed the "geometrical" information on the face like vertices, triangles structure and, optionally the texture node with the information about the texture image to apply and the texture coordinates.

We define some FAE requirements to take into account for the system we have developed:

- The Facial Animation Engine has to be independent of the shape and size of the model to be animated.
- It has to be able to load the geometric description of the facial model from a VRML file.
- The animation rules for each FAP has to be able to be defined whatever is the geometric wire-frame model to be animated.
- The system has to be able to create a full animation between different predefined expressions.
- The animation technique to be used has to be the Interpolation between predefined points.
- The system has to be able to generate the animation choosing among three interpolation techniques: lineal, polynomial and spline.

3.SOME RESULTS, CONCLUSIONS AND FUTURE WORK

We define an implement a low cost FAE using real image capturing, without any marker on face.

In the figure 4, we can see a mobile phone with the image general with our system, is similar in size to one SMS and with a MMS we can send also the texture. In any case, after the initial process only we need to send the changing points in animation. We are working hard to define a application in Java specially dedicate to mobile requirements to using the owns mobile camera we can do all the process.

Figure 4. Mobile phone with the image general with our system

We have described an implementation of a Face Animation Engine compliant with MPEG-4 capable of animating a generic facial wire-frame by providing the usual geometric parameters together with some semantic information on the wire-frame.

Many aspects are still to be improved. Implementation of the movements affecting other parts of the face. The portability of the system is also another of the tasks to be exploded. The ability of the system to be exported to Internet and to the mobile networks will multiply the number of applications to be used.

4.ACKNOWLEDGEMENTS

This paper is subsidized by the projects:

- HUMODAN IST-2001-32202
- CICYT TIC2001-0931.
- TIC2002-10743-E.

Many thanks to all members of the Computer Graphics and Vision Group at UIB.

5.REFERENCES

[Bli01a] J. Blinn. A generalization of algebraic surface drawing. ACM Transaction on Graphics. v. 1, n. 3, pp. 235-256, July, 1982.

[Jör01a] Jörn Ostermann. Face Animation in MPEG-4. 2002. AT&T Labs-Research, Middletown, New Jersey.

[Lee01a] W. S. Lee; M. Escher, G. Sannier; N. M. Thalmann. MPEG-4 compatible faces from orthogonal photos. IEEE Computer Animation. Geneva. May 1999.

[Lee01b] Y. C. Lee; D. Terzoupoulos; K. Waters. Realistic face modelling for animation. Siggraph Proceedings, 1996, pp. 55-62.

[Par01a] F. I. Parke. A parametric model for human faces. Thesis, University of UTA, SALT Lake City – USA. UTEC – CSC-75-047,1997.

[Pie01a] L. Piegl; W. Tiller. The NURBS book. 2.ed. Springer, 1997.

[Snh01a] SNHC, Information Technology. Version of 13. Nov. 1998, ISO/IEC JTC1/SC29/WG11 N2502a. Atlantic City, Oct. 1998.

[Ven01a] C. Venter; D. Weber. Three-dimensional reconstruction of faces. 1999 IEEE Africon 5th Conference, September 1999, v. 1.

POSTERS proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

A flexible and modular architecture for object space NC-Machining simulation

Aitor Moreno[(1)] Carlos Toro[(1)] Iosu Arizkuren[(1)] Álvaro Segura[(1)] Jorge Posada[(1)]
{amoreno, ctoro, iarizkuren, asegura, jposada}@vicomtech.es
Marcelino Novo [(2)] mnovo@aotek.es
Juanjo Falcón [(3)] jjfalcon@somesi.com
Nieves Alcaín[(4)] nalcain@alecop.es

[(1)] **VICOMTech.** Paseo Mikeletegui 57. 20009 San Sebastián (Spain)
[(2)] **FAGOR AUTOMATION S. Coop.** Bº San Andrés, 19 - Apdo.144 - 20500 Mondragón (Spain)
[(3)] **SOME Sistemas Informáticos S.L.** Avda. Navarra s/n (oficina 10). 20500 Mondragón (Spain)
[(4)] **Alecop, S. Coop.** Loramendi, 11, Apto. 81 - 20500 Mondragón (Spain)

ABSTRACT

The NC machining processes based on computer graphics is a significant part of modern Computer Integrated Manufacturing (CIM). Through simulation, it is possible to test the correctness of the NC tool paths without the need of machining actual physical parts, with the corresponding reduction of time and costs. In this paper, we present a flexible and modular architecture that describes a generic object-based NC machining simulation system. This architecture has been tested using several object-based simulation approaches, and could be successfully integrated in commercial simulation systems. This work also covers other proposed architectures found in the literature.

Keywords

NC-Machining, Verification, Material Removal, Architecture, Extensible, Flexible, Solid Representation, BREP, Octree, SIMUMEK.

1. INTRODUCTION

The simulation and verification of NC machining processes based on computer graphics is nowadays a significant part of modern Computer Integrated Manufacturing (CIM). Through previous simulation, it is possible to test the correctness of the NC tool paths without the need of machining physical parts, with the corresponding reduction of time and costs.

2. SCOPE AND PREVIOUS WORK

The NC machining simulation using computer graphics techniques can be traced back to the classical works of Anderson and Van Hook [VAN86]. Since then, several researchers have proposed different approaches, classified in two main groups: *(i)* image-based methods [SAI91] and *(ii)* object-based (or non image-based) [CRO00], [SIM04].

3. PROPOSED ARCHITECTURE

Posters proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

High Level Modules Definition

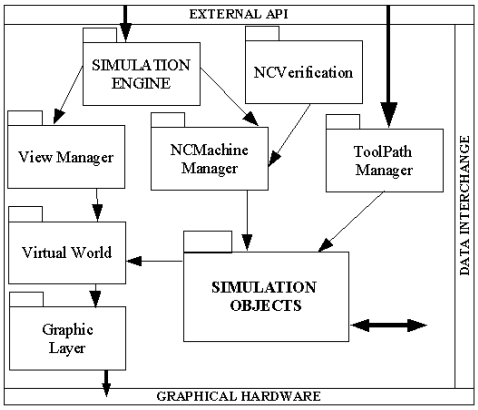


Figure 1. Architecture Main Modules

Simulation Engine	Manages the user input/output process.
View Manager	Responsible of the management of the views.
Machine Manager	Responsible of the management of the NC machines
Simulation Objects	Static objects, lights, primitives, 2D forms, etc.
Virtual World.	Objects, lights and views organized in an scenegraph
Graphical Layer	Wrapper to a graphical API like OpenGL or DirectX.
ToolPath Manager	Path for the tool

Table 1. Architecture Main Modules explanation

Input-Output Modules

External API	External interface to the user
Graphical Hardware	Rendering of the virtual world with a graphical API
Data Interchange	Load or save the geometrical information

Table 2. I/O module explanation

Simulation Objects Module

Static Objects.	Clamps, holders, arms, etc (environment)
Light.	Light source used to light the virtual world
Stock.	Stock definition
Spatial partitioning	Stock partitioning based
Tool	The tool is affected by a movement producing a sweep volume to be subtracted from the stock
Movements	Movements as outputs from the parsing process of the G-Code
Sweep.	A sweep volume of the moving tool
Material Removal.	Boolean subtraction between the Stock and the sweep volume.
Low Level Verification Module	Collision detection, feedback forces, heat transfer

Table 3. I/O Simulation object s module explanation

Simulation Engine Module

Optimized Control	Real time control
Finite State Machine	Alters the actual simulation state if necessary
Time Based Simulation	Handling of time related events

Table 4. I/O Simulation engine s module explanation

4. VALIDATION

The validation of the proposed architecture has been accomplished through the development of some heterogeneous prototypes.

a) **B-Rep Based Geometric Representation with Internal Spatial Partitioning-** Implemented in ACIS and OpenCascade

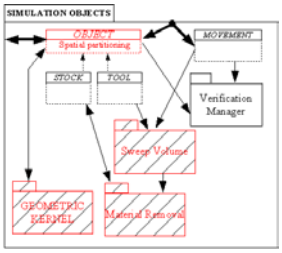


Figure 2. OpenCASCADE and ACIS Supported Object Modules

b) **CAD-Supported Octrees Geometric Representation with Implicit Spatial Partition-** Implemented in AutoCAD ObjectARX.

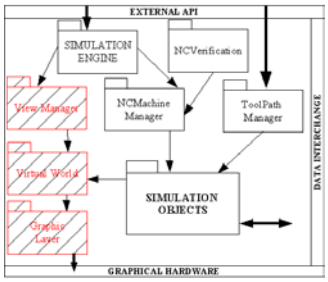


Figure 3. AutoCAD Supported Main Modules

c) **Polygonal Based Geometric Representation with Internal Spatial Partitioning**– implemented in the SIMUMEK project [SIM00].

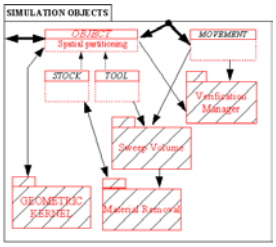


Figure 4. SIMUMEK Supported Object Modules

The SIMUMEK kernel provides an internal spatial partitioning for the stock object. The spatial partition is not based in a hierarchical representation. Its definition is closer to a hash table with a quick access to the inner elements. A nice feature is the adaptative partitioning allowing reactions at simulation time.

5. CONCLUSIONS

In this paper a flexible and modular architecture for non image based NC-Machining simulation was presented. Some of the benefits of the presented schema e.g. implementation flexibility, modularity and time saving in the development stage were proved by the presentation of various test applications. The test models were implemented in different schemas, combining different solid model representations (Octrees, B-Rep based) and different API's as well in order to show the feasibility of the architecture.

6. REFERENCES

[CRO00] Crossman, J.; Yoon, D., "A cutter motion simulation system" Transaction of the Society for Design and Process Science 2000 pp. 25-35.

[KAR00] Karunakaran K.P, Pawan K., Mondani P.K, Gupta N, Shanmuganathan P.V, Garg S, "Octree-Based Volumetric NC Simulation System" 2000.

[SAI91] T. Saito, T. Takahashi, "NC Machining with G-buffer Methods", Conputer Graphics, Volume 24, Number 4, July 1991. pag. 207

[SIM04] SIMUMEK Project Homepage http://www.vicomtech.es/ingles/html/proyectos/index_proyecto20.html

[VAN86] Tim Van Hook, "Real Time shaded NC Milling Display", SIGGraph86, Volume 20, Number 4, pag. 15-20

Automatically Generating 3-D Image Imagined from Drawing Lines

Satoru Morita
Faculty of Engineering, Yamaguchi University
2557 Tokiwadai Ube 755-8611 Japan

Abstract

In the paper, we automatically create a three-dimensional image using the occluding contours that exist in a viewed image. We propose a new method to generate the surface based on the rules needed to reconstruct the stable surface known in the psychology field from the occluding contours. The method is useful because it can be used as a modeler in computer graphics.

1 Introduction

Human can imagine three-dimensional images based on drawing lines. This paper describes a method to make a three-dimensional image from drawing lines. The important rules are discussed to imagine three-dimensional image from the contour in the psychological field[1][2]. A method which qualitatively estimates three-dimensional objects from a silhouette contour[3] is also discussed. Recently, drawing lines are automatically detected from the picture[4]. Three-dimensional objects are detected from drawing lines[5][6]. The interface used to easily make three-dimensional object from drawing lines is the main purpose. The purpose of this study is to make a three-dimensional object from drawing lines. Such a goal cannot be achieved without the use of a model.

On the other hand, the method to reconstruct three-dimensional object from a contour using the general cylindrical model and superquadrics is discussed in the field of computer vision[7][8]. The shape strongly depends on the function and model.

As well, the detection of an object motion in order to make three-dimensional animation based on a cartoon has been attempted[9]. The deformation of three-dimensional surface is estimated in the case that the basic three-dimensional model exists. In this paper, we reconstruct a shape using the rules that the contour normal vector is perpendicular to the viewing direction on points on the contour and the surface is smooth on the point in which a contour does not exist.

2 Rules to reconstruct the stable surface from the occluding contour

The following important rules are necessary to make a three-dimensional image based on occluding contours[1][2]. In general, a contour exists on the gray boundary in the viewing image. The gray boundary exists on the color boundary between neighboring pixels in the viewing image. So the contour exists on the color boundary between neighboring pixels in the viewing image. Though the color boundary does not exist on the object surface, the contour is sometimes caused by occluding the surface in the viewing image. We call this the occluding contour. We make a three-dimensional image based on the occluding contours.

The stable surface necessary to cause the occluding contours satisfies the following rules.

- The normal direction of the tangential line on a point existing on an occluding contour is perpendicular to the viewing direction.
- If an occluding contour does not exist in this position, the surface is smooth.

We generate the curved surface which satisfies the stable surface rules to generate a three-dimensional smooth surface.

3 Reconstructing a three-dimensional image based on an occluding contour

A surface is made from a circle contour using the knowledge that the contour shape is a circle. If the shape satisfies the stable surface rules, it is a sphere whose radius is the same as that of the circle. At first, all points existing on a circle are connected to the circle's center. Next, we define the depth which is the radius of the circle. Thus, the sphere surface is reconstructed from the circle contour. There are a point that have the constant distance from all points existing on a circle. The point is the circle center. The estimated sphere radius is the same as the circle radius in the case that the contour is the circle.

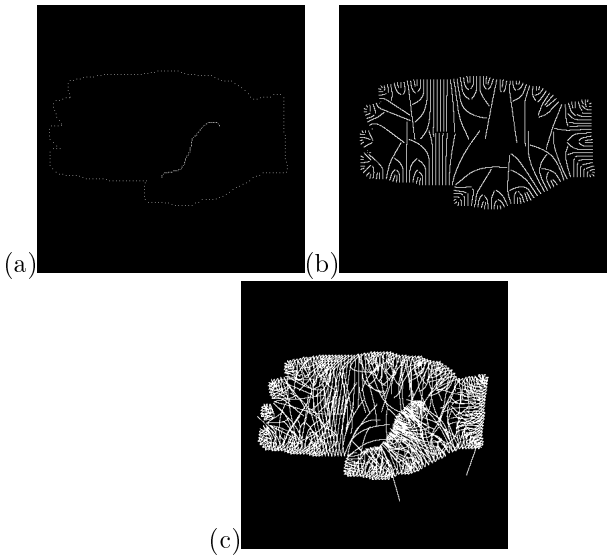


Figure 1: Three-dimensional image made from the occluding contours of a hand.(a) A hand contour. (b) The process by which points existing on the silhouette contour move. (c) All generated surfaces.

Moreover, the points move to the perpendicular direction for the tangential line on contour with the velocity based on the curvature. If the curvature is big, the velocity is fast. If the curvature is small, the velocity is slow. $\vec{x}(s,t)$ shows the position on the contour, with the contour represented using parameter $0 < s < S$. The points existing on the contour move according to the time t. $\vec{n}(s,t)$ shows the normal direction of the contour position, and κ shows the curvature of the position on the contour. The points on the contour move with the velocity based on curvature. The diffusion velocity F of the equation is based on the curvature, and the direction of movement on the position $(x_t(s,t), y_t(s,t))$ is perpendicular to the tangential line in the points existing on the contour.

$$(x_t(s,t), y_t(s,t)) = F(\kappa) \cdot \vec{n}(s,t) \tag{1}$$

Where,

$$F(\kappa) = 1 - 0.25\kappa \tag{2}$$

$$n(\vec{s},t) = \frac{(y_s(s,t), -x_s(s,t)))}{(x_s(s,t)^2 + y_s(s,t)^2)^{\frac{1}{2}}} \tag{3}$$

$$\kappa = \frac{y_{ss}(s,t)x_s(s,t) - x_{ss}(s,t)y_s(s,t)}{(x_s(s,t)^2 + y_s(s,t)^2)^{\frac{3}{2}}} \tag{4}$$

The points on the contour move to the position where they contact to the other point. The moving distance is measured. The depth in the contour is defined as the moving distance. The point on the contour moves Δxy and the depth changes Δz. The moving line is divided into max lines. We calculated the jth points in max number. The max lines are defined by the depth $\theta_b = \frac{\pi}{2}\frac{j+1}{max}$ $\theta_a = \frac{\pi}{2}\frac{j}{max}$.

$$\Delta xy(j) = rcos(\theta_b) - rcos(\theta_a) \tag{5}$$

$$\Delta z(j) = rsin(\theta_b) - rsin(\theta_a) \tag{6}$$

$(j = 0 \cdots max - 1)$ Thus, the contour depth is defined.

4 Making a three-dimensional image based on occluding contours

We show the algorithm used to generate the three-dimensional image.

- The occluding contours are detected from a viewing image.
- A direction for an occluding contour is assigned.
- All points existing on the occluding contours move in a direction perpendicular to the tangential line in the points existing on the occluding contour, and the points stop when it contacts other point.
- The distance in which all points existing on the occluding contour move is calculated.
- Depth is calculated using the distance moved and we define the depth such that the shape is a sphere.

Figure 1(a) is a hand contour, and Figure 1(b) shows the process by which points existing on the silhouette contour move. Figure 1(c) shows an estimation of all surfaces of the hand. All surfaces of the hand surfaces are generated. We confirmed that we can generate three-dimensional image using the occluding contour from this result.

References

[1] D. D. Hoffman, "Visual intelligence," NORTON, 1998.

[2] J. J. Koenderink, "What does the occluding contour tell us about solid shape?," Perception, 13, pp. 321-330, 1984.

[3] D. D. Hoffman, "Parts of recognition," Cognition, 18, pp. 65-96, 1984.

[4] B. Gooch and A. Gooch, "Non-Photorealistic Rendering, A K Peters, 2001.

[5] T. Igarashi, S. Matsuoka and H. Tanaka, "Teddy: A Sketching Interface for 3D Freeform Design," proc. of SIGGRAPH, pp. 409-416,1999.

[6] R. C. Zeleznik, K. P. Herndon, and J. F. Heghes, "SKETCH: An interface for sketching 3D scenes," proc. of SIGGRAPH 96, pp. 163-170, 1996.

[7] A. P. Pentland, B. Horowitz and S. Sclaroff, "Closed-form Solusions for physically based shape modeling and recognition,"IEEE Trans. on PAMI, 13(7), pp. 715-729, 1991.

[8] A. Vidmar and F. Solina, "Reconstruction of superquadrics from 2d contour," Theoretical Fundation of Computer Vision, Mathematical Research, vol. 69, pp. 227-240, 1992.

[9] P. Rademacher, "View-dependent Gerometry," proc. of SIGGRAPH, pp. 439-446, 1999.

Different levels of interaction in a Virtual Environment

D. Oyarzun
Edutainment and Graphical UI Department
VICOMTech Research Centre
Paseo Mikeletegi,57
20009 San Sebastian, Spain
doyarzun@vicomtech.es

A. Ortiz
Edutainment and Graphical UI Department
VICOMTech Research Centre
Paseo Mikeletegi,57
20009 San Sebastian, Spain
aortiz@vicomtech.es

ABSTRACT

The widespread use of electronic devices in the daily tasks has motivated the research in user interfaces. The conversational user interfaces using virtual characters are starting to be widely used in order to improve the human-computer communication, due to the illusion of having a conversation with a real human. Our approach is based on having an avatar as the main interaction element in the virtual environment. Thanks to the virtual characters we can "connect" all the components in the system, such as user, VE and avatars, and having a high level of interaction. In one hand, the user can interact with the avatar and with the environment. In the other hand, the avatars can interact with the user or behave in an autonomous mode, interacting between them. Having this kind of interaction implies the development of animation techniques, path finding, collision detection, human behaviours, etc. The techniques implemented for achieving the different levels of interaction are also explained in this paper.

Keywords

User interfaces, animation techniques, Virtual characters, path-finding, collision detection

1. INTRODUCTION

In order to have a natural communication with electronic devices, it is obvious that the interaction with the user interfaces should be totally different to the actual desktop paradigm based on windows, icons, mouse and pointers (WIMP Paradigm), which almost have not been changed since it appears in 1984. In this paper we explore the possibility of improving the interaction using virtual characters.

Projects as [Oli00a, Eur04a, Mar03a] have been focused their work in having a realistic avatar, which can help or represent the user in the virtual environment.

Our approach is based in this premise but extended to having different levels of interaction.

In section 2, our way of understanding the different levels of interaction, which should be included in a virtual environment, are explained. The necessary techniques, including the explanation of our facial and body animation engine and the static and dynamic collision detection, are explained in section 3. In section 4, the prototype, which consists in a virtual museum with two avatars integrated, is explained. The avatars can interact with the user or the environment, or act in an autonomous mode, interacting between them.

Conference proceedings ***ISBN 80-903100-8-7***
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. INTERACTION IN THE VIRTUAL ENVIRONMENT

Our approach is based on having an avatar as the main interaction element in the system. In this way, the system is going to be made up of avatars, the virtual environment and the user. Thanks to the virtual characters we can "connect" all the components in the system, obtaining the following levels of interaction:

Between the user and the avatar: If the user asks about some information related with the environment, the avatar will react in a natural way, with verbal and corporal language.

Between the user and the environment: The user is able to move through the environment freely, interacting with the objects and navigating through the rooms.

Between avatar and environment: As the user, the avatar is able to move through the environment freely, interacting with the objects and navigating through the rooms, avoiding the walls or other avatars which are in the environment

Between avatars: The avatar can adopt several roles in the environment. By means of behavioural rules, their behaviour when they are not interacting with the user can be defined.

3. ACHIEVING THE INTERACTION

Every levels of interaction involve the development of techniques that can carry out them.

Animation engine

A facial animation engine has been developed using advanced morphing techniques [Ale00a] and individual deformations and transformations. The animations are defined using a VHML authoring tool [Car04a].

The body animation engine uses VRML for predefined animations as walk or run, and inverse kinematics for real-time generated movements as pointing [Wel89a].

Collision detection and automatic computation of new trajectories

Other fundamental module in the application is the collision manager. Without it, the avatar could move in the virtual world without taking the relative position of walls, objects or other virtual characters into account. We have developed an algorithm in order to solve it.

In a first step, the algorithm calculates a trajectory between two desired points avoiding the walls; it is done abstracting the virtual world into cells and connections between them. Each cell is an area that has not static obstacles and where the avatar can walk in straight line without colliding. These cells are used in order to create a graph to perform a heuristic search. Then, using the A* algorithm, the shortest path between the two points is obtained.

In a second step, the possible collisions with dynamic elements, as other avatars, are detected. First, the algorithm tries avoid a detected collision stopping one avatar; then, if the collision persist, we shirk between them by means of calculus the internal tangent of the two circles that abstracts the avatars. This step is periodically done on an efficient way. Having the speed of the avatars into account, the minimum time that they would take on meet another time is calculated, and it will be the next time that the step will be executed.

Finally, the definitive trajectory is computed taking into account the positions of the dynamic and static obstacles, because, due to the changes in the trajectory in order to avoid the dynamic obstacles, a new collision with a static obstacle could be caused.

4. DESCRIPTION OF THE PROTOTYPE

In order to test the techniques developed, a prototype has been developed.

Our prototype is a virtual museum, where two virtual assistants are integrated, and the user can navigate and interact with the environment, with the objects and with the avatars. The virtual world and the avatar's body are made in VRML and the avatar's head is a polygonal model saved in the Alias/Wavefront format (*.obj).

While nonuser is interacting with them, they start to interact by means of some predefined behaviour rules. When they meet, they salute with more or less effusively depending on the time that has been since the last salute.

The user has the possibility of asking for information about the pictures of the museum to one avatar. In that moment, the selected avatar will get out of its autonomous condition, it will come near of the picture that the user is asking for, and it will start the explanation about the picture.

5. REFERENCES

[Ale00a] Alexa, M., Behr, J., and Müller, W. The Morph Node. Proc. Web3d/VRML 2000, Monterey, CA., pp.29-34. 2000

[Car04a] Carretero M.P., Oyarzun D., Aizpurua I., and Ortiz A. Animación Facial y Corporal de Avatares 3D a partir de la Edición e Interpretación de Lenguajes de Marcas. Congreso Español de Informática Gráfica CEIG,pp.139-150.2004

[Eur04a] Eurocitizen: a web-based game of knowledge &collaboration for Europe. Retrieved on October 2004; from: http://eurocitizen.mls.gr

[Mar03a] Marques Soares, J., Horain, P., and Bideau, A. Sharing and immersing applications in a 3D virtual inhabited world. Laval Virtual 5th virtual reality international conference. pp. 27-31.2003

[Oli00a] Oliveira, J.C., Shen, X. and Georganas, N.D. Collaborative Virtual Environment for Industrial Training and e-Commerce", Proc. Workshop on Application of VirtualReality Technologies for Future Telecommunication Systems, IEEE Globecom. 2000

[Wel89a] Welman, C. Inverse kinematics and geometric constraints for articulated figure manipulation B. Sc. Simon Fraser University, 1989

Ultrasonic Diffraction Tomography: The Experimental Result

C. Pintavirooj† K. Jaruwongrungsee† W. Withayachumnankul‡
K. Hamamoto* S. Daochai**

†Research Center for Communication and Information Technology (ReCCIT), and
Department of Electronics, Faculty of Engineering,
‡Department of Information Technology,
King Mongkut's Institute of Technology Ladkrabang, Thailand.
*Department of Information Media Technology.
School of IT. Tokai University. Japan
**Institute of Science and Technology for Research and Development
Mahidol University at Salaya, Thailand

ABSTRACT

Diffraction tomography is a technique for imaging with acoustic fields. It takes advantage of the linearization process of the non-linear wave equation describing wave propagation in heterogeneous media. When the scattering effect is weak, one can invoke the Born or Rytov approximation and thus derive the generalized Fourier Slice Theorem to reconstruct the cross-section of the insonified object. Although diffraction tomography is a promising technology for medical application as it provides a quantitative ultrasonic image, its realization toward medical use is still far-to-go, this may be due to the complexity of the hardware involved. In this research we investigate a potential use of diffraction tomography for medical application by using a delicate-designed ultrasonic computerized tomographic system. The result of experiment investigation of diffraction tomography is very promising.

Keywords

ultrasound, tomography, diffraction.

1. INTRODUCTION

Ultrasound has potentially many important technological applications. These include medical imaging [Jof90a], nondestructive testing [Bol89a], and robotic vision [Bol89a]. The advantages of ultrasound imaging offered over more conventional imaging are numerous. They include the relatively low health hazard of non-ionizing, low power of sources, its ability to image physiological properties of a tissue or organ, and the likely cost competitiveness of the imaging equipment. The difficulty with ultrasound imaging is associated problem with performing object reconstruction. Because ultrasound imaging experience significant attenuation, scattering problem and diffraction, standard tomographic reconstruction schemes are not readily applicable and are of only limited usefulness. In an attempt to overcome this problem, many approaches to ultrasound imaging has been investigated in the recent past[Kak88a]. Ultrasonic Diffraction tomography is a technique for inverting the differential wave equation governing interaction between the insonifying field and the scattering medium. This paper investigate the experimental result of ultrasonic diffraction tomography.

WSCG 2005 POSTERS proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. ULTRASONIC DIFFRACTION TOMOGRAPHY

When an object is insonified with a plane wave and the detector is located at y = l_0. as shown in Fig. 1, the propagation of acoustic waves and the detected scatter field can be modeled with the Helmholtz's wave equation.[sla95a]

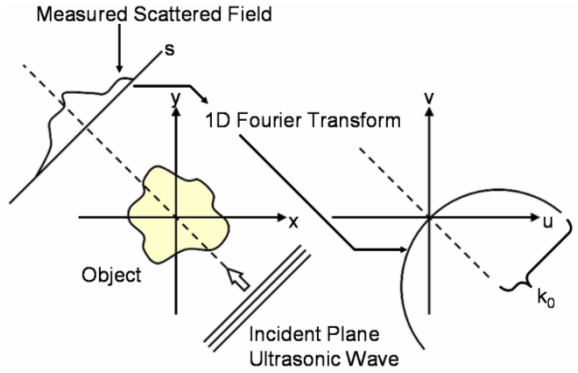


Figure 1. Fourier Diffraction Theorem.

which finally leads to the generalized Fourier slice theory defined by :

$$\Phi_s(\omega, l_0) = \frac{j}{2\sqrt{k_o^2 - \alpha^2}} e^{j\sqrt{k_0^2 - \alpha^2} l_o} \cdot$$

$$O(\alpha, \sqrt{k_o^2 - \alpha^2} - k_0) \qquad (1)$$

where

$\Phi_s(\omega, l_0)$ is the Fourier transform of the scatter field detected by the detector.

$O(\alpha, \sqrt{k_o^2 - \alpha^2} - k_0)$ is fourier transform of the object evaluated along the circular arc of radius $k_0 = 2\pi/\lambda$, which is the wave number.

By illuminating an object in many different directions and measuring the diffracted projection data, one can fill up the Fourier space with the samples of the Fourier transform of the object over ensemble of circular arcs and then reconstruct the object by Fourier inversion**.**

3. EXPERIMENTAL RESULTS

The scattered field is measured by placing the specimen on the rotating platform and insonifying the object with ultrasonic plane-wave transducer. The needle-shaped ultrasonic receiver is linearly scanned to pick up the transmitted pulse. After one complete linear scan, the specimen is rotated to the next angle and the process is repeated. Figure 8 shows phantoms and the reconstruction results. The phantoms are made from the gelatin with different refractive index.

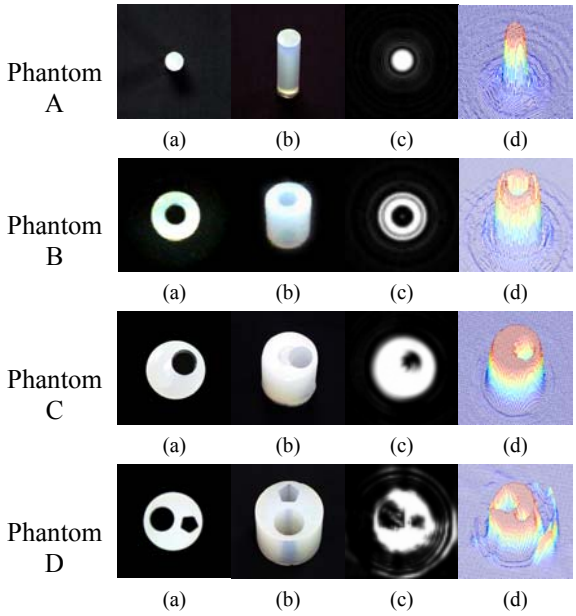


Figure 2. Phantom and Reconstructed Results

(a) Top view (b) side view (c) Reconstructed Results. (d) mesh of (c). (Number of projection: 8, scanning interval: 1mm. Refractive index of phantom A: 1.055 diameter: 1 cm. MSE : 13.81, B: 1.033 diameter: 3.5 cm. MSE : 11.09, C: 1.012 diameter: 3.5 cm. MSE : 13.29, D: 1.017 diameter: 3.5 cm. MSE : 26.64)

4. DISCUSSION AND CONCLUSION

We investigated the quantitative ultrasonic imaging using the diffraction tomography. The experiment results indicate that the diffraction tomography method provides quantitatively accurate imaging. Despites its promising technique, diffraction tomography is subjected to various limitations, which include artifacts due to diffraction effects in strong inhomogenous media.

5. REFERENCES

[Kak88a] Kak, A. C. and Slaney, M., "*Princicles of Computerized Tomographic Imaging*", IEEE Press, NY, 1988.

[Sla95a] Slaney, M., "Imaging with Diffraction Tomography,", Ph.D. Thesis, Purdue University, 1895.

[Jof90a] Jofre, L., Hasley, M. S., Broquetas, A., Reyes, E., Ferrando, and M. Elias-fuste, "Medical Imagingwith Microwave tomographic Scanner," *IEEE. Trans. Biomed. Eng.,* vol.37, no. 3, pp. 303-312, 1990.

[Bol89a]Bolomey, J., -C., "Recent Eurepean Developments in Active Microwave Imaging for Industrial, Scientific, and Medical Application," *IEEE. Trans. Microwave Theory Tech*., vol. 37., no. 12, pp. 2109-2117, 1989.

Simulation of blur in stereoscopic image synthesis for virtual reality

Benoît Piranda François de Sorbier Didier Arquès
SISAR Team, University of Marne La Vallée
5, boulevard Descartes, Champs sur Marne
77454, Marne La Vallée, France
piranda@univ-mlv.fr fdesorbi@univ-mlv.fr arques@univ-mlv.fr

ABSTRACT

This paper proposes a method permitting to generate synthesized images including blurred area in real time in order to help stereoscopic perception in virtual reality systems. At any time of the process, the combined knowledge of the scenario and the position of the user in front of the screen allows to select automatically the important zone of the scene. The elements associated to the clear zone catch the attention of the user while the blurred areas avoid an excessive eyestrain.

First, we present several methods permitting to simulate blur effects produced by a thin lens in image synthesis. Then we develop our work on the adaptation of the previous methods in the stereoscopic images context, illustrated by results generated in our own virtual reality system. Finally, we propose different solutions for the treatment of the interdependence between scenario and interactivity in image synthesis animations.

Keywords

real time image synthesis, simulation of blur, virtual reality, scenario, interactivity, stereoscopic images.

1. INTRODUCTION

One of the fundamental characteristics of a virtual reality system [Fuc01] is the immersion of the user in a virtual environment. In a visualization context, the using of stereoscopic image synthesis permits to recreate three-dimensional environments by mentally fusing two images of a single scene.

On the one hand, when the two stereoscopic images are locally different (parallax), it implies an excessive eyestrain or even the impossibility for the user to reconstruct the 3D scene. On the other hand, all objets displayed on screen have potentially the same importance. The real time synthesized images produce a perfectly sharp image in all its surface : the user will cast a glance over the screen.

The main goal of our work consists in both simulating, in real time, the perception of stereoscopic images including depth of field in order to catch the attention of the user in the important areas of the scene, and reducing excessive eyestrain.

In this paper, we first present the main methods permitting to generate stereoscopic images and to simulate the depth of field. Then we detail our new method adapting one of these existing algorithms to the context of the stereoscopic images production applied to virtual reality. Finally, we propose some rules permitting to select important elements in virtual scene.

WSCG 2005 POSTERS proceedings, ISBN 80-903 100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. STEREOSCOPY AND BLUR

2.1. Computation of stereoscopic images

The stereoscopic images principle consists in associating an image of the same scene with different points of view for each eye. Thanks to a pair of filter glasses, the user merges these two images in order to create the relief effect.

2.2. Adding blur

2.2.1 Notion of circle of confusion

A lens (simulating human eye) creates an image of an object whose position p is function of the lens' focal length. If we place a projection plane at the position p the image will be sharp else a stain will appear on this plane at the origin of the blur effects.

This stain will be defined as the circle of confusion [Pot82].

2.2.2 Blur calculation methods

We classify the methods of synthesized images production that take into account the depth of field into two categories. First the preprocessing methods that calculate a high resolution image [Coo84] [Woo99], and next the post-processing methods that locally degrade images in order to add blur [Pot82].

One of this methods, that we are using for our work and mainly used in video games, permits to add blur in real time synthesized images [Can01] [Eva01]. It consists in calculating a set of sub-resolutions images (mipmaps). They are all placed orthogonally and linearly in front of the point of view, completely covering the screen. The farther the images are from the original one (in the foreground), the lower their resolution is. Finally, for each pixel, the size of the circle of confusion deduced from the depth map permits to know which plane to display.

3. BLUR IN VIRTUAL REALITY

In the stereoscopic images context, with the addition of the notion of convergence plane that is different from the focus plane, we have to improve the method presented in the last paragraph. Our method consists in placing a set of blurred images before and behind the focus plane. The areas that may cause eyestrain, because of the shifting between two stereoscopic images, can become blurred and as a consequence reduce their interest for the user.

If the images are placed linearly, the transition between two images could be perceptible to the user. So we have organized our images with an exponential function first to reduce the distance between the images close to the focus plane and second to increase the space between remote images. An example of generated image is presented figure 1.

4. THE INTEREST AREA

4.1. Imposing the focus plane position

To direct the attention of the user we position the focus plane in a zone that we define as the interest area. If the user behind the virtual reality screen is inactive we use a scenario to automatically modify

Figure 1 : image with depth of field simulation

Figure 2 : Interactive positioning example

the position of the focus plane during the animation and highlight objects of our scene that have more importance than others. For a given time *t*, the scenario also modifies the geometrical and radiometrical characteristics of the objects to get the visual result we are looking for.

4.2. Interactive positioning

The degree of freedom of the user can be increased by using his position and orientation in the virtual scene to find the position of the focus plane. For example, the unblurred object is the biggest one in front of the user. So as to place the focus plane, we use morphological parameters of elements in the image, in this way the user will pay attention, for example, to the nearest object in the screen (figure 2).

5. CONCLUSION

In this paper we have presented a new solution for real time stereoscopic images synthesized in the context of virtual reality using blur. By highlighting areas that we have defined as important for the user and by reducing interest of areas where the parallax between two stereoscopic images is too high, this method permits to decrease the user's eyestrain.

The next part of this work will be to empirically validate this solution with sets of users in our virtual reality system.

6. REFERENCES

[Can01] R. Cant, N. Chia and D. Al-Dabass. New anti-aliasing an depth of field techniques for games. The second International Conference on Intelligent Games and Simulation: 117-122, 01.

[Coo84] R. L. Cook, T. Porter, and L. Carpenter, Distributed Ray Tracing. CG, 18(3):137-145, 84.

[Eva01] A. Evans. Four Tricks for Fast Blurring in Software and Hardware: 3, 01.

[Fuc01] Philippe Fuchs, Guillaume Moreau, and Jean-Paul Papin. Le traité de la réalité virtuelle. Les Presses de l'Ecole des Mines, 01.

[Pot82] M. Potmesil and I. Chakravarty. Synthetic image generation with a lens and aperture Camera Model. ACM TOG, 1(2) : 85-108 ,82.

[Woo99] M. Woo, J. Neider and T. Davis. OpenGL 1.2. CampusPress : 460, 99.

A Low Cost Structured Light System

Mário L. L. Reiss
São Paulo State University
Rua Roberto Simonsen, 305,
Pres. Prudente, SP, Brazil
19060-900, Presidente
Prudente, São Paulo
mreiss@prudente.unesp.br

Antonio M. G. Tommaselli
São Paulo State University Rua
Roberto Simonsen, 305, Pres.
Prudente, SP, Brazil
19060-900, Presidente
Prudente, São Paulo
tomaseli@prudente.unesp.br

Christiane N. C. Kokubum
São Paulo State University, Rua
Roberto Simonsen, 305, Pres.
Prudente, SP, Brazil
19060-900, Presidente
Prudente, São Paulo
kokubum@prudente.unesp.br

ABSTRACT
The aim of this paper is to present the current development status of a low cost system for surface reconstruction with structured light. The acquisition system is composed of a single off-the-shelf digital camera and a pattern projector. A pattern codification strategy was developed to allow the pattern recognition automatically and a calibration methodology ensures the determination of the direction vector of each pattern. The experiments indicated that an accuracy of 0.5mm in depth could be achieved for typical applications.

Keywords
Structured light, 3D imaging, calibration, low cost.

1. INTRODUCTION
Reconstruction with structured light system is still a topic of interest in Photogrammetry and Computational Vision due to its effectiveness for close range object reconstruction. Basically, 3D reconstruction with structured light consists in projecting a light pattern (line, grid, or shape) onto an object, and recovering the shape of object using the projection geometry. The technique can be based on the projection of a spot (a laser beam) or a frame pattern, which is reflected by the object and recorded by a digital camera [Bal82a]. Three key problems can be mentioned: geometry of the projection; identification and precise location of the projected pattern in the digital image and; system calibration [Dun89a]. There are several different approaches to solve the issues concerning the camera and projector calibration, reconstruction, codification of patterns and precise measurement of patterns, and some of them can be found in: [Tom98a], [Bat98a], [Zha02a], [Sal04a].

A system called 3DScanSL (3D Scanner by Structured Light) has been developed based on an off-the-shelf digital camera, a pattern projector and software components, mainly for system calibration processing and surface reconstruction. This paper presents the current status of system development. A brief derivation of the mathematical models, the prototype configuration and experiments with real data will be presented in this paper.

2. THE 3DScanSL
The 3D reconstruction system is composed by an acquisition system (Figure 1), a calibration plate, and computer software implemented in C++ language. All the software components were in house developed and the prototype was in house projected and mounted by an external private company (AVR Instrumental).

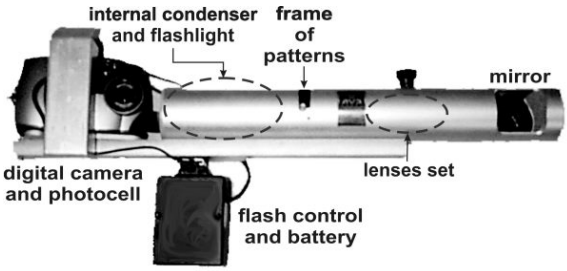


Figure 1. The system prototype.

The developed prototype uses a Kodak DX 3500 digital camera, with a maximum resolution of 1800x1200 pixels, fixed focal length (38 mm) and a pixel size of 19.44 μm (35mm equiv.). The digital camera is tightly attached to the projector ensuring the geometric conditions for reconstruction that are mathematically modeled. The reconstruction is achieved through the parametric equation of the projected ray (pattern) combined with the well-known collinearity equations [Tom98a]:

$$X_{i,t} = X_{\mathbf{p}} + \lambda_{i,t} \cdot l_t,\ Y_{i,t} = Y_{\mathbf{p}} + \lambda_{i,t} \cdot m_t,\ Z_{i,t} = Z_{\mathbf{p}} + \lambda_{i,t} \cdot n_t \quad (1)$$

Considering the projective equations:

$$x_{i,t} = -f\frac{X_{\mathbf{p}} + \lambda_{i,t} \cdot l_t}{Z_{\mathbf{p}} + \lambda_{i,t} \cdot n_t},\ y_{i,t} = -f\frac{Y_{\mathbf{p}} + \lambda_{i,t} \cdot m_t}{Z_{\mathbf{p}} + \lambda_{i,t} \cdot n_t} \quad (2)$$

where: $x_{i,t}, y_{i,t}$ are the image coordinates of the t^{th} projected point in the i^{th} image; f is the camera focal length; $\lambda_{i,t}$ is a scale factor; $X_{i,t}, Y_{i,t}, Z_{i,t}$ are 3D

coordinates of the projected point in the camera reference system; (X_P, Y_P, Z_P) are the coordinates of projector center, and (l_t, m_t, n_t) the direction cosines of the projected ray.

The first step is the projector calibration aiming the determination of the projector center and the direction cosines of each projected pattern. To achieve this, the patterns are projected onto a flat surface; assuming $Z_{i,t}$ as a constant, Equations (1) can be rewritten as:

$$X_{i,t} = l_t \cdot (Z_{i,t} + Z_P) - X_P,\ Y_{i,t} = m_t \cdot (Z_{i,t} + Z_P) - Y_P \quad (3)$$

All the elements of Equation (3) are related to the camera reference system. Using this equation, the coordinates of the projector center and the components of the direction vectors can be estimated in an iterative simultaneous adjustment of observations and parameters. The XY coordinates of the projected points in the camera reference system are considered as pseudo-observations, whilst $Z_{i,t}$ is set as a constant. At least two images with a minimum of four projected points over the reference plane at different distances are required. The components l_t and m_t of the direction vector are computed considering n_t as negative unit value (-1); a normalization can be carried out later for each direction vector.

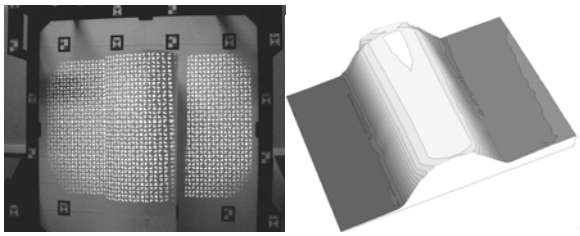
A pattern codification strategy was developed to allow the recognition of the patterns automatically (Fig. 2)

Figure 2. Projected patterns: (a) The five primitive patterns; (b) An example of a group of nine patterns from a collected image.

The 5 primitives L-shaped (Fig. 2) targets were grouped to enable recognition of incomplete patterns and quality control. A matrix of 54x36 targets (1944 targets) was generated and reproduced by photographic process in a slide that is insert in the pattern projector.

3. EXPERIMENTS WITH REAL DATA

In order to access the accuracy and the potential of the methodology, the prototype was tested with simulated and real data. The camera was firstly calibrated using a bundle adjustment with convergent cameras. After, the projector parameters were computed using the developed methodology. Several tests were performed to evaluate the accuracy of 3D reconstruction achieved by the system. The experiments demonstrated that 0.5mm in depth and 0.2mm in XY are the typical accuracies with the current configuration. The prototype was then tested with a solid cylindrical object, resulting in the 3D model shown in Fig 3.



(a) (b)

Figure 3. (a) Acquired image and, (b) 3D surface model of a cylinder.

4. CONCLUSIONS

The presented results show that an accuracy within 0.5mm can be achieved and this is considered suitable for the intended applications, with a cost less than US$ 3000. The system performance and accuracies can be significantly improved by introducing a high resolution digital camera and using subpixel point measurement technique, like least squares template matching.

5. ACKNOWLEDGMENTS

This work has been supported by Fapesp (Fundação de Amparo à Pesquisa do Estado de São Paulo).

6. REFERENCES

[Bal82a] Ballard, D. H. and Brown, C. M. *Computer Vision***.** Prentice-Hall, New Jersey, USA, 1982.

[Dun89a] Dunn et al. Measuring the Area and Volume of the Human Body with Structured Light. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6), 1989.

[Tom98a] Tommaselli, A. M. G. Geometric Accuracy of a Structured Light System. *In: International Archives of Photogrammetry and Remote Sensing*, Cambridge, UK, 32, pp.313-319, 1998.

[Bat98a] Batlle, J. et al. Recent Progress in Coded Structured Light as a Technique to Solve the Correspondence Problem: A Survey. *Pattern Recognition*, 31, pp.963-982, 1998.

[Sal04a] Salvi, J. et al. Pattern Codification Strategies in Strucutured Light Systems, *Pattern Recognition*, 37, pp.827-849, 2004.

[Zha02a] Zhang, G.; Wei, Z. A Novel Calibration Approach to Structured Light 3D Vision Inspection. *Optics & Laser Tecnology*, 34, pp.373-380, 2002.

Layer-based Decompositions of Polyhedra

A. J. Rueda, F. Martinez and F. R. Feito
Escuela Politécnica Superior
Campus Universitario Las Lagunillas
23071 Jaén, Spain
{ajrueda|ffeito}@ujaen.es

ABSTRACT

This work describes a decomposition scheme for polyhedra called Layer-based decomposition. This decomposition can be computed in a efficient way for any kind of polyhedron, and has interesting applications in several geometric problems, like Boolean operation computation, point-in-polyhedron inclusion test, 3D location and ray-scene intersection computation.

Keywords

3D decompositions, Geometric Algorithms.

1. INTRODUCTION

Decomposition techniques are extensively used in the areas of Geometric Modelling, Computational Geometry and Computer Graphics as a useful tool in the description of objects and the design of simple algorithms for non-trivial problems. Triangulations, decomposition into trapezoids or convex subpolygons, and triangle or quad mesh generation are well-known decompositions in 2D [Ber00, Ber95, Goo97]. Tetrahedra decomposition and tetrahedra or hexahedra mesh generation are similar examples in 3D [Ber00, Ber95].

The layer-based decomposition of polygons was already presented in previous works [Fei99], describing several of its applications: Boolean operations computation, point-in-polygon inclusion test or location test [Riv00, Rue02]. Recently, this decomposition has been generalized to polyhedra [Rue02, Rue04], showing interesting properties and applications like 3D Boolean operations computation, point-in-polyhedron inclusion test, ray-scene intersection test or hidden surface removal.

Most part of this work describes the computation of the layer-based decomposition of polyhedra, illustrating this decomposition with a full example. In the last part we give some ideas beyond the decomposition of polyhedra into extended layers, which is currently an active area of research.

WSCG 2005 POSTERS proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. DECOMPOSITION PROCESS

For the sake of simplicity we will only consider polyhedra consisted of triangular faces, although this is not a strong restriction as any polyhedron can be converted to this type just by triangulating its faces. We start by choosing an arbitrary point called *origin*. This point will determine the size and number of layers of the decomposition. From this point we make a full covering of the polyhedron by building tetrahedra between the origin and every triangular face of the polyhedron.

The layer-based decomposition of a polyhedron is based on the study of the *subordination relations* [Fei99, Riv00] between the tetrahedra from this covering:

Definition 1 *Given two tetrahedra $s = \triangle OP_0P_1P_2$ and $t = \triangle OQ_0Q_1Q_2$, we say s* is subordinated to *t, or t* dominates *s ($s \lhd t$), if $s = t$ or in the case $s \neq t$ there exists a point inside the face of s opposite to O which belongs to the interior of t.*

It is important to highlight that the subordination relation is only defined for tetrahedra determined by a common point origin and two non-intersecting triangular faces.

We use a primary data structure to store all the tetrahedra of the covering, and for each tetrahedron a secondary data structure storing references to every tetrahedra that dominates it. Building this data structure requires testing subordination between every two tetrahedra, with a $O(n^2 \log n)$ time cost if a binary search

tree is used for the secondary data structure. The layer decomposition algorithm works as follows:

1. *Generate all the tetrahedra between the point origin O and the triangular faces of the polyhedron.*
2. *Build primary and secondary data structures computing subordination between the tetrahedra. Initialize layer counter i to 1.*
3. *Check the primary data structure for tetrahedra with no dominating tetrahedra (secondary data structure with no elements).*
4. *Add these tetrahedra to layer L_i.*
5. *Delete these tetrahedra from the primary data structure and all their occurrences in the second data structures of the rest of tetrahedra.*
6. *Increment i and return to step 3 until there are no tetrahedra in the primary data structure.*

The second phase of the decomposition algorithm also has a $O(n^2 \log n)$ time cost with an efficient implementation of the secondary data structure. The result is a set of layers $L(P,O) = \{L_1, L_2, ..., L_k\}$ which constitutes the layer-based decomposition of the polyhedron P respect to the point origin O. See [Fei99, Rue02, Rue04] for a thorough and more formal introduction to the concept of layer, its properties and decomposition algorithms.

3. APPLICATIONS

The layer-based decomposition has been successfully applied to several geometric problems:

- *Boolean operations computation.* Starting with the layer-based decomposition of two objects, it is possible to compute the result of a Boolean operation combining the tetrahedra of both decompositions following a different criterion depending on the operation [Riv00].
- *Point-in-polyhedron inclusion test.* This test is based on testing the point against each layer in descending order. Once the test succeeds for a given layer, the rest of tetrahedra can be discarded without testing [Rue02, Rue04].
- *Location test.* The point-in-polyhedron inclusion test can be generalized to a point location test if the layer-based decomposition is applied to a scene with several polyhedra [Rue02, Rue04].
- *Ray-scene intersection test.* A common layer-based decomposition of the polyhedra from a scene taking the observer as origin can be useful for determining the polyhedron intersected by a ray starting from the observer [Rue04].
- *Hidden surface removal.* The previous layer structure can also be used for the implementation of the Painter's algorithm. Rendering the tetrahedra of each layer following an ascending order gives a correct final visual result of the scene from the point of view of an observer placed at the origin of the layer decomposition [Rue04].

4. ACKNOWLEDGEMENTS

This work has been partially granted by the Ministry of Science and Technology of Spain and the European Union by means of the ERDF funds, under the research projects TIC-2001-2099-C03-03 and TIN-2004-06326-C03-03.

6. REFERENCES

[Ber95] Bern, M., Eppstein, D. Mesh generation and optimal triangulation. *Computing in Euclidean Geometry 2nd ed.*, pp. 189-196, World Scientific, 1995.

[Ber00] Bern, M., Plassmann, P. Mesh generation. *Handbook of Computational Geometry*, pp. 291-332, North Holland, 2000.

[Fei99] Feito, F. R., Rivero, M. L., Rueda, A. J. Boolean representation of general planar polygons. *Proc. WSCG'99*, pp. 87-92, 1999.

[Goo97] Goodman, J. E., O'Rourke, J. (eds.) *Handbook of Discrete and Computational Geometry*, CRC Press, 1997.

[Riv00] Rivero, M. L., Feito, F. R. Boolean operations on general planar polygons. *Computer & Graphics*, 24, pp. 881-896, 2000.

[Rue02] Rueda, A. J., Feito, F. R., Rivero, M. L. A triangle-based representation for polygons and its applications, *Computer & Graphics*, 26, pp. 805-814, 2002.

[Rue04] Rueda, A. J. *Representación de Objetos Gráficos mediante Capas y sus Aplicaciones*. PhD Thesis, Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, 2004 (in spanish).

CyberSession: A New Proposition for E-Learning in Collaborative Virtual Environments

Javad Sadeghi
Computer Engineering Department
Iran University of Science and Technology, Tehran, Iran
javad@iust.ac.ir

Amir Aavani
Computer Engineering Department
Iran University of Science and Technology, Tehran, Iran
a_aavani@iust.ac.ir

Mohsen Sharifi
Computer Engineering Department
Iran University of Science and Technology, Tehran, Iran
msharifi@iust.ac.ir

ABSTRACT

E-Learning in fact is another type of learning process through the integration of technology and is perceived as learning via a web browser, over the web or an intranet network. E-Learning in the context of 3D virtual environments promises better performance in the light of new trends for 3D environments among users.

This paper presents the design and first steps to the implementation of a collaborative environment representing course materials through web3D technologies. It highlights the unique ability to run in low bandwidth by simulating user input and transforming work forces into a knowledge resource. In our framework, nicknamed *CyberSession*, users can switch between 2D and 3D environments to support synchronous training, asynchronous training and collaborative training systems.

Keywords

E-Learning, Collaborative Virtual Environments, Social awareness, Avatars

1. INTRODUCTION

Today, with promotion of computer based training and electronic learning methods, many universities and research institutes invest in virtual classrooms. E-Learning is a popular term and has various extensions such as Online learning, Web-based training (WBT), Web-based learning and technology based training (TBT) and Computer-based training (CBT).

As innovation accelerates, training programs must become more flexible and less time consuming. However 70% of people never finish WBT systems because of the lack of a social contact. Using collaborative WBT system we have access to an updateable real-time interactive content alongside virtual social awareness [Gut 95]. Collaborative interactive learning is a memorable experience. Questions are answered immediately in lively discussions. Learning in groups makes the course fun and ensures a supportive atmosphere [Bla04].

In this paper we present our proposition nicknamed CyberSession to overcome the current problems of multimedia classes and bring better performance to e-learning methods and discuss the lessons we have learned from our experiences.

WSCG 2005 POSTERS proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic

2. CYBERSESSION TESTBED

In CyberSession there exists only a Server program that is responsible for transformation of messages between client programs. According to the current design, the server program must be executed on a single computer and it has the following capabilities:

Specifications of the system

1) Definition of minimum online users
2) The ability of changing the teacher (without changing people from behind of computer)
3) The ability of saving a session and re-execution of it without presence of teacher
4) The ability of holding an exam (in this case each workstation works stand-alone)
5) It is to a high extent independent of the program being taught
6) Support of UDP or TCP communication
7) Sharing one or multiple files

CyberSession is implemented as a case study on a LAN and its performance is tested. In CyberSession instead of sending all of the display to computers of class members, only events of keyboard and mouse is transferred to computers of the cyber class.

3. CYBERSESSION ARCHITECTURE IN COLLABORATIVE VIRTUAL ENVIRONMNETS (CVE)

In our new architecture we have conceived all useful appliances in a cyber class room. The functionalities that are supported in this way are: **Communication between attendees, Social awareness, Avatar simulator, Billboard simulator, Class administration** [Bou01].

The architecture of our system is dictated in Figure 1. The teacher's computer will send two different messages to the Server that are indicated in the diagram with (a) and (b).

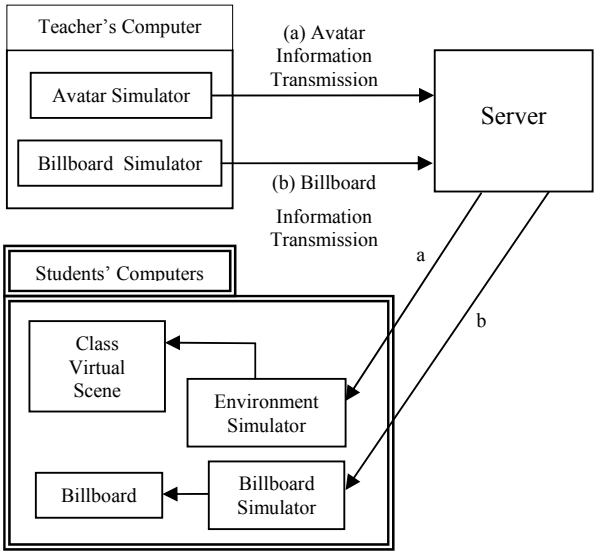


Figure 1. CyberSession Architecture

The (a) type messages contain the teacher avatar movements while in (b) type messages information related to billboard changes will be sent to the Server. The students will receive two messages (a) and (b) from the server that are the messages from the teacher's computer.

4. EVALUATION

E-Learning methods on Internet generally reduce training costs dramatically and increase the productivity of trainers since they have access to learning contents.

Main benefits of our approach

1) Low price in contrast with current costs of multimedia classes.
2) Ability of practical representation of course with all details.
3) Ability of requesting any student at any point of the course representation to continue the job.
4) Possibility for practical evaluation of students.
5) Students may ask questions/interact with the teacher's permission.
6) Actual physical location for course representation is unnecessary.
7) Ability of streaming the voices of the teacher to students, student to teacher and student to students with suitable performance depending on available bandwidth.
8) Ability of saving and re-representing course sessions for review or re-use with low volume, low price and without loss of quality.
9) Ability of offline representation of saved courses (without need for the presence of teacher).

5. CONCLUSION

In our current system users will attend in a fully interactive class and with low cost of communication can access to huge amount of information.

Our main future trend is implementation of the CVE version of CyberSession with the design reported in this paper. We are also in the middle way of negotiation with Blaxxun Technologies GmbH to use their Blaxxun Platform as a container for CyberSession project.

6. REFERENCES

[Bou01] Bouras, C., Hornig, G., Triantafillou, V. and Tsiatsos, T. *"Architectures Supporting e-Learning Through Collaborative Virtual Environments: The Case of INVITE",* In Proceedings of IEEE International Conference on Advanced Learning Technologies-ICALT 2001, Madison, Wisconsin, USA, August 6-8, 2001, pp. 13-16.

[Gut95] Gutwin, C., Stark, G. and Greenberg, S. *"Support for Workspace Awareness in Educational Groupware"*. Pro ACM Conference on Computer Supported Collaborative Learning, Indiana University, Bloomington, Indiana, USA October 17-20, 1995, pp 147-156.

[Bla04] Blaxxun Technologies GmbH, http://www.blaxxun.com, 2004

Modeling with Rational Bézier Solids

Martin Samuelcik
Department of Applied Informatics,
Faculty of Mathemathics, Physics and Informatics,
Mlynska dolina
84248, Bratislava, Slovakia
samuelcik@fmph.uniba.sk

ABSTRACT

In this paper we present a technique for modeling solids based on the rational trivariate Bézier expressions. These solids are defined by analytical expression. For modeling purposes we focus on rotational, transitional and twisted solids. Final visualization is then done by approximation of solids by net of points and by boundary evaluation of solid. We also present practical output of our visualization algorithm.

Keywords

rational Bézier solids, rotational, transitional, twisted, solids visualization

1. INTRODUCTION

The trivariate solids are not very popular in geometric modeling because of high degree of freedom. Rather they were used for free-form deformations and modeling [Sed86]. But, for some part of geometric modeling, they are useful. It is very easy to map 3D textures on such solids, also some properties can be attached to the points of solid with trivariate functions. Some types of solids can be used for deformation simulation because of their transformation invariance. Also, there exist works considering approximations of these solids by subdivision schemes [Mcd02]. In this paper we focus on one group of trivariate solids, rational Bézier solids.

2. RATIONAL BÉZIER SOLIDS

The rational Bézier tetrahedral is defined with a degree *n*, tetrahedral domain *ABCD*, control net of points, and for each point one real number (weight). Control net with weights can be written in following way:

$$V_{\mathbf{i}} \in E^3; w_{\mathbf{i}} \in R$$
$$\mathbf{i} = (i, j, k, l); |\mathbf{i}| = i + j + k + l = n; i.j, k, l \geq 0$$

Let us have point *U* from the domain and let $\mathbf{u}=(u,v,w,t); u+v+w+t=1$ are barycentric coordinates of point *U* ($U=uA+vB+wC+tD$) with respect to *ABCD*. Point of rational Bézier tetrahedra $RB^n(\mathbf{u})$ can be defined using analytical expression:

$$RB^n(\mathbf{u}) = \frac{\sum_{|\mathbf{i}|=n} w_{\mathbf{i}} V_{\mathbf{i}} B_{\mathbf{i}}^n(\mathbf{u})}{\sum_{|\mathbf{i}|=n} w_{\mathbf{i}} B_{\mathbf{i}}^n(\mathbf{u})}; B_{\mathbf{i}}^n(\mathbf{u}) = \frac{n!}{i!\,j!k!l!} u^i v^j w^k t^l$$

Rational Bézier tensor solid is defined with three degrees *n*, *m*, *o*, box domain *ABCDEFGH* and a control net of points and for each point a real number (weight):

$$V_{\mathbf{i}} \in E^3; w_{\mathbf{i}} \in R$$
$$\mathbf{i} = (i, j, k); n \geq i \geq 0; m \geq j \geq 0; o \geq k \geq 0$$

Assume that we have point *U* from domain and let **u** are coordinates of *U* with respect to *ABCDEFGH*, so $\mathbf{u}=(u,v,w); 0 \leq u,v,w \leq 1$. Now we can define point of Bézier tensor solid $RB^{n,m,o}(\mathbf{u})$ with analytical expression:

$$RB^{n,m,o}(\mathbf{u}) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{k=0}^{o} w_{(i,j,k)} V_{(i,j,k)} B_i^n(u) B_j^m(v) B_k^o(w)}{\sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{k=0}^{o} w_{(i,j,k)} B_i^n(u) B_j^m(v) B_k^o(w)}$$

where

$$B_i^n(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i}$$

are Bernstein polynomials.

WSCG 2005 POSTERS proceedings, ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

3. MODELING

Translational solids

Let us have a rational Bézier patch with given control points $P_{(i,j)}$ and weights $s_{(i,j)}$, where $i=0,..,n;j=0,\ldots,m$ and rational Bézier curve with control points Q_i and weights r_i, where $i=0,\ldots,o$. The curve and patch must correspond to each other, they share their first vertices, so $P_{(0,0)}=Q_0$. Then translational rational Bézier tensor solid has degrees (n,m,o) and following control points and weights:

$$V_{(i,j,k)} = P_{(i,j)} + Q_k - Q_0; w_{(i,j,k)} = s_{(i,j)} * r_k$$
$$0 \le i \le n; \qquad 0 \le j \le m; \qquad 0 \le k \le o$$

Rotational solids

Lets have rational Bézier curve with control points $Q_i=[Qx_i, Qy_i, Qz_i]$ and weights r_i, where $i=0,\ldots,o$. Then rotational rational Bézier tensor solid has degrees $(2,2,o)$ and the following control points and weights:

$$V_{(0,0,i)} = [Qx_i, Qy_i, Qz_i]; w_{(0,0,i)} = r_i;$$
$$V_{(0,1,i)} = [Qx_i + Qy_i, Qy_i - Qx_i, Qz_i]; w_{(0,1,i)} = r_i;$$
$$V_{(1,0,i)} = [Qx_i - Qy_i, Qx_i + Qy_i, Qz_i]; w_{(1,0,i)} = r_i;$$
$$V_{(0,2,i)} = [Qy_i, -Qx_i, Qz_i]; w_{(0,2,i)} = 2r_i;$$
$$V_{(1,1,i)} = [0,0,Qz_i]; w_{(1,1,i)} = r_i;$$
$$V_{(2,0,i)} = [-Qy_i, Qx_i, Qz_i]; w_{(2,0,i)} = 2r_i;$$
$$V_{(1,2,i)} = [Qy_i - Qx_i, -Qx_i - Qy_i, Qz_i]; w_{(1,2,i)} = 2r_i;$$
$$V_{(2,1,i)} = [-Qx_i - Qy_i, Qx_i - Qy_i, Qz_i]; w_{(2,1,i)} = 2r_i;$$
$$V_{(2,2,i)} = [-Qx_i, -Qy_i, Qz_i]; w_{(2,2,i)} = 4r_i;$$

where $i=0,\ldots,o$.

Twisted solids

The twisted solid is like a translational solid, but when we are translating patch along the curve, on each level we rotate translated control points around given axis by the given angle. So assume that we have rational Bézier patch given by control points $P_{(i,j)}$ and weights $s_{(i,j)}$, where $i=0,..,n;j=0,\ldots,m$ and a rational Bézier curve with control points Q_i and weights r_i, where $i=0,\ldots,o$. We will be rotating each level patch around z-axis by angle α. The control points and weights of resulting rational Bézier tensor patch with degrees (n,m,o) are:

$$V_{(i,j,k)} = (P_{(i,j)} + Q_k - Q_0) * \begin{pmatrix} \cos(k\alpha) & \sin(k\alpha) & 0 \\ -\sin(k\alpha) & \cos(k\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$w_{(i,j,k)} = s_{(i,j)} * r_k$$
$$0 \le i \le n; \qquad 0 \le j \le m; \qquad 0 \le k \le o$$

4. RESULTS & CONCLUSION

Based on rational Bézier solids and its approximation by point nets, we visualized some solids modeled using described ways. We have presented trivariate solids based on Bézier notion for splines and prepared basic modeling tools for creating such solids

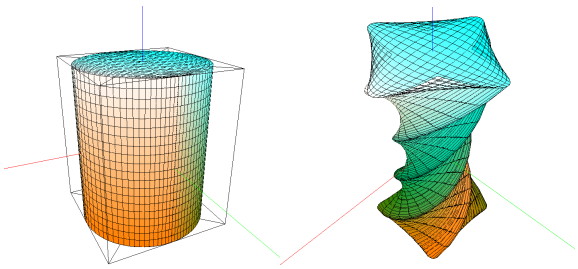


Figure 1. Transitional and twisted rational Bézier tensor solids.

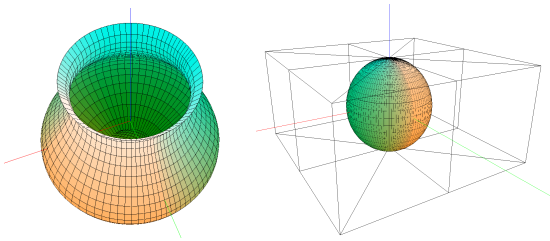


Figure 2. Rotational rational Bézier tensor solids

5. ACKNOWLEDGMENTS

This work has been supported by the VEGA grant Virtual Environments for WWW No. 1/0174/03.

6. REFERENCES

[Far93] Farin G. Curves and Surfaces for Computer-Aided Geometric Design - A Practical Guide. ISBN 0-12-249052-6, Academic Press, Boston, 1993.

[Pie97] Piegl L., Tiller W. The NURBS Book. ISBN 3-540-61545-8, Springer-Verlag, New York, 1997.

[Joy99] Joy K., Duchaineau, M., Boundary determination for trivariate solids, Proc. of Pacific Graphics '99, pp 82-91, Oct 1999.

[Mcd02] McDonnell, K., Qin, H., Dynamic Sculpting and Animation of Free-form Subdivision Solids, The Visual Computer 2002, pp. 81-96, 2002

[Sed86] Sederberg, T. W. and Parry, S. R., Free-Form Deformation of Solid Geometric Models, Proceedings of SIGGRAPH ‘86, Computer Graphics 20, 4 (August 1986), 151-159.

[Ogl04] OpenGL webpage. http://www.opengl.org

A Tracking Algorithm for Rigid Point-Based Marker Models

Bernd Schwald
Computer Graphics Center (ZGDV), Department Visual Computing
Fraunhoferstraße 5
D-64283 Darmstadt, Germany
bernd.schwald@zgdv.de

ABSTRACT

Tracking of objects and persons by using a stereo camera system setup and markers is applied since several years in Virtual and Augmented Reality or other applications. In especial infrared tracking systems are getting increasingly popular, because of their precision and robustness. Often, markers for such systems are spherical and therefore not distinguishable, making the consideration of geometric constellations of several markers necessary to identify objects and to determine their transformations. This paper presents an algorithm for tracking rigid constellations of markers, which can be adapted to the needs of corresponding applications.

Keywords

Tracking, Marker, Stereo Camera System, Geometric Constellation, Infrared

1. INTRODUCTION

Using punctual objects as markers for optical tracking is very typical for infrared stereo camera systems. Those markers are passive retrore ective spheres or active infrared diodes with a big angle of radiation. Such markers have the advantage, that they can be chosen quite small, but still their position can be reconstructed precisely in a large interaction volume.

To determine translation and rotation of objects, it is necessary to consider rigid constellations of markers, here called models, respectively Multi Point Models (MPMs). One important task is to achieve a high reliability of the detection of MPMs by regarding properties like distances between points and translation and rotation of the MPM in the previous frame. Nevertheless a limitation to some criteria for the detection of MPMs is necessary, in order to prevent extensive computations causing too much latency between capturing images and returning the calculated transformations.

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

In the following, the existence of a set of $m \in \mathbb{N}$ 3D points $\mathcal{X}_W = \{X_W^1, \ldots, X_W^m\}$, reconstructed in one frame from synchronously captured camera images, is supposed. The 3D points are the result of a segmentation on the camera images and a triangulation thanks to a calibrated system, as described e.g. in [Har97], which are then processed by the tracking algorithm. The 3D points are the world coordinates of markers and therefore more or less accurate, depending on the used system. Furthermore, disturbing points not representing a really existing marker may be elements of this set.

2. MULTI POINT MODEL

A Multi Point Model (MPM) consists of parameters

$\mathcal{X}_M$ the set $\{X_M^1, \ldots, X_M^n\}$ of 3D points, $n \geq 3$, and
id a unique identi er for the model.

These parameters are de ned once, when the model is initialised. From the set $\mathcal{X}_M$ of 3D model points the set $\mathcal{D}_M = \{d_M^{jk} \in \mathbb{R} : 1 \leq j \leq n-1, j < k \leq n\}$ of $n_d = \frac{n(n-1)}{2}$ distances between model points X_M^i can be calculated once and stored, whereas $d_M^{jk} = |X_M^j - X_M^k|$. Furthermore the model consists of parameters

$\mathcal{K}$ the set of correspondences between reconstructed points X_W^j from the previous frame and model points X_M^i,
R the model rotation from the previous frame and
T the model translation from the previous frame.

Normally, these parameters change during tracking of the model in each frame.

3. DETECTION OF MODELS

For detecting a MPM, the model points X_M^i, representing the real markers, have to be known. In simple cases their coordinates can be measured by hand or by just placing the model once in the interaction volume of the tracking system. In more complex cases a special procedure, allowing movements of the model during the learning procedure [Sch04], can be applied.

The fact, that 3D points are reconstructed not perfectly, is important when comparing distances and point coordinates. To take this reconstruction error into account, a distance tolerance $\delta > 0$ is de ned, expressing that a distance $d_{ij} = |X_i - X_j|$ between two points X_i, X_j is regarded as equal to a given distance $d \in \mathbb{R}$, if $|d_{ij} - d| < \delta$. For the world and the model coordinate system have the same scaling, the distance tolerance δ is not de ned depending on a certain coordinate system, but is the same for all coordinate systems.

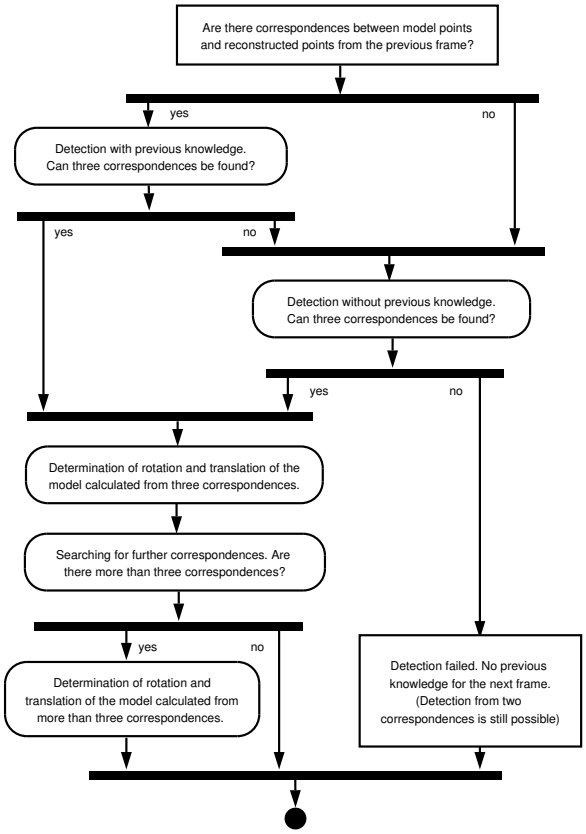


Figure 1. Tracking algorithm for MPMs based on both detection with and without previous knowledge from last frame.

Starting point for the detection of a MPM in the present frame is the set $\mathcal{X}_W = \{X_W^1, \ldots, X_W^m\}$ of reconstructed points in world coordinates in one frame. Two different methods are applied: one method is based on previous knowledge from the last frame, the other does not use previous knowledge. In both cases the rst step consists in trying to identify the MPM by nding three correspondences between model points X_M^i and reconstructed points X_W^j. Therefore the distances between the reconstructed points are calculated and compared with the distances stored in the model.

The detection method without previous knowledge is mainly based on comparing distances between world points with distances between model points. In order to make this more ef cient, knowledge form the previous frame is added in the second method. Nevertheless the rst method is important, because previous knowledge is not always available. Both detection methods, are combined in one tracking algorithm, see Figure 1.

4. DISCUSSION AND RESULTS

The presented tracking algorithm is implemented as part of an infrared tracking system, allowing to track several models. To evaluate the processing time, needed by the tracking algorithm, up to four models with two times three, four and nine markers were tracked. The tracking software was running on an Athlon XP 1800+ with 1 GB RAM. The models were moved within and out of the interaction volume, such that the number of actually tracked models varied between 0 and 4 at a given time. Table 1 shows the results from ≈ 700 measurements. The processing time Δ, includes image processing, 3D reconstruction from two camera images and the presented tracking algorithm.

#models	#tracked frames	average Δ in ms	average #markers	approx. #dist.
0	85	4.8	1.4	0
1	267	5.9	9.9	45
2	148	6.2	14.3	91
3	117	6.6	17.0	136
4	104	7.0	18.9	171

Table 1. Between 0 and 4 models were tracked in ≈ 700 measurements. Δ is the processing time for the corresponding number of tracked models.

Obviously, the processing time increases with the number of models and the number of reconstructed points, but ≈ 7.0 ms is an acceptable value for a setup with four models consisting of 19 markers. In especial, the processing time is not increasing the same way as the number of distances between reconstructed points does.

5. REFERENCES

[Har97] R. Hartley and P. Sturm, Triangulation, Computer Vision and Image Understanding 68 (1997), no. 2, 146 157.

[Sch04] B. Schwald and P. Figueiredo, Learning of Rigid Point-Based Marker Models for Tracking with Stereo Camera Systems, 1. Workshop der GI VR/AR (Chemnitz), 2004, pp. 23 34.

Real-Time Simulation of Autonomous Vehicles on Planet-Sized Continuous LOD Terrains

Selcuk SUMENGEN
Computer Graphics Laboratory
Sabanci University
selcuk@su.sabanciuniv.edu

Selim BALCISOY
Computer Graphics Laboratory
Sabanci University
balcisoy@sabanciuniv.edu

ABSTRACT

Real-Time visualization of interactive simulation environments using large datasets of height fields became feasible using current off the shelf graphics hardware. Our approach provides continuous level of detail rendering of high detailed, planet sized terrains using restricted quad-trees without re-sampling data points. The presented method preserves the original planet coordinate frame of the data gathered from the Mars Orbiter Laser Altimeter with 128 samples/degree resolution for vehicle simulation purposes. Furthermore the algorithm avoids discontinuities at the block boundaries occurring at latitudes.

Keywords

Terrain visualization, quad-tree, planet size simulation.

1. INTRODUCTION

Visualization of large scale planet-sized terrain datasets with interactive frame rate is a complex problem. The system should coupe with Gigatriangle sets. Moreover rendering of terrain is not sufficient in most of the application areas. The system should take care of other tasks such as simulation of the environment, vehicle navigation, collision detection and response. This paper presents a complete system which focuses on interactive terrain visualization and simulation of vehicles on it.

The presented algorithm preserves the co-ordinate system of the elevation data. Thus there is no necessity for further pre-processing. Moreover we preserve geometric continuity of the planet surface and therefore prevent cracks on the surface at every region of the planet.

2. TERRAIN VISUALIZATION

Terrain engine renders elevation data with several detail levels determined by surface slope and distance from the camera. The presented algorithm uses quadtree data structure to store and access data points as well as performing frustum culling and triangle tessellation.

Restricted Quad-Tree (RQT) [Paj98] algorithm limits the level of detail difference between the neighbourhood nodes to one level. This approach reduces polygon count while preserving terrain detail without degrading the overall performance.

Previous approach on visualization of planet uses PBDAM [Cig03] (Planet-Sized Batched Dynamic Adaptive Meshes) structures to maintain geometric continuity, which presents mapping of a planet to a cube having six faces as a square patches. This approach requires pre-processing for re-sampling the elevation data which is an additional cost and a handicap for dynamic update of height fields.

Generating Restricted Quad-Tree

Arrangement of the elevation data is a rectangle form. However, planets have 180 latitudes and 360 longitudes after cylindrical projection, and therefore their elevation data sets can not be used with quad tree structures directly.

We position the rectangular sample frame into a square by overlapping centres. Restricted Quad Tree will be formed using square mapping. Some of the points in the square will remain unmapped after placing the rectangle frame into a square frame. Marked regions in the Figure 1 are formed by these

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

unmapped points but they will also remain unreferenced after branching the tree because available coordinates for viewing the terrain is inside the latitude 0 – 360 and longitude 0 – 180 and these nodes of the tree will never collapse with a little modification on branching algorithm. This method will degrade the memory usage and prevent unnecessary calculations.

Restricted Quad Tree is always successful for preserving geometric continuity inside the square; however there might be discontinuities on the boundaries such as transition from longitude 0 directly to longitude 359 will cause cracks caused by level of detail changes. This is avoided with the limitation of the maximum viewing area to 180 longitudes.

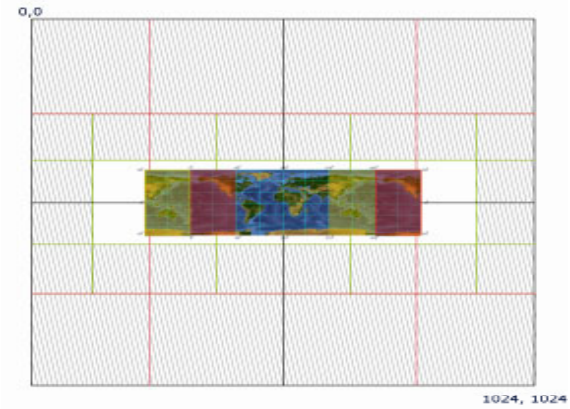


Figure 1: Square Region Mapping: Mapping of elevation data to a square after cloning rightmost and leftmost regions to preserve geometric continuity at boundaries.

Branching of Restricted Quad-Tree

Elevation data samples with resolution of N can be placed in a square sized (1024 x N + 1) x (1024 x N + 1), so every 1° x 1° region corresponds to nodes at level 10. The corresponding quad tree structure will be branched to level 10 and level 10 is preserved as base level. This produces enough number of triangles while viewing half of the planet as a whole. Expansion of nodes that are below level 10 should be determined after checking viewing volume, viewers distance from the planet, and slope of the corresponding planet surface.

While expanding the nodes to several levels (Figure 2 a-b-c), LOD difference between the neighbouring regions should be concerned to avoid discontinuities and cracks. If there is a LOD difference between the neighbours the difference must not be more than one (Figure 2 c-d). This constraint can be satisfied easily using any recursive branching algorithm. Another requirement is marking the triangle on the neighbouring side of lower level region for split operation from its bisector if there is a level of detail difference. Reason of this split operation can be seen from figure 2 where the regions with higher level of detail have extra vertices on shared sides with neighbours having lower level of detail. If these vertices on the same side are on different planes, there will be cracks on the surface. These cracks are prevented by splitting the corresponding triangle of lower level region from its bisector. After this operation smooth transitions of different level of details achieved (Figure 2f).

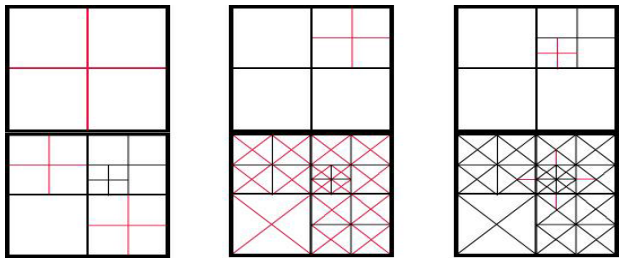


Figure 2: Branching Restricted Quad-Tree: from left to right: a-) 1. LOD b-) 2.LOD, c-) 3.LOD, d-) Level Refinement, e-) Triangulatization, f-) Connecting levels

3. Results

We used NASA's 128 sample/degree data set for Mars surface (1.93GB) and 4 sample/degree data set for Moon surface. The Figure 3 is a snapshot from real-time rendering of Mars surface with 400 ground and air vehicles with 60 frames/seconds.

Figure 3: Mars Simulation Environment

4. REFERENCES

[Cig03] Cignoni, P., F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio and R. Scopigno, "Planet--sized batched dynamic adaptive meshes (P-BDAM).", In Proceedings IEEE Visualization, Conference, Seattle USA, October 2003.

[Paj98] Pajarola, R., "Large scale terrain visualization using the restricted quadtree triangulation.", IEEE Visualization'98, North Carolina, 1998

A flexible approach to non-homogeneous texture generation

Francesca Taponecco
TUD GRIS
Department of Computer Science
Fraunhoferstrasse 5
64283, Darmstadt, Germany
ftapone@gris.informatik.tu-darmstadt.de

Thomas Rieger
TUD GRIS
Department of Computer Science
Fraunhoferstrasse 5
64283, Darmstadt, Germany
rieger@gris.informatik.tu-darmstadt.de

ABSTRACT

Due to the numerous applications, textures reserve significant interest in many scientific fields. Several recent works concentrate in the analysis and synthesis of textures, and they still are a very active and actual area of research in Computer Graphics. In this work, we propose a pixel-based technique, which allows the production of smooth textures and a precise patterns' visualization. Textures' patterns may move or vary along given directions or may progressively change their appearance. We implemented our algorithm incorporating much functionality in a general framework; this framework offers the possibility to generate non-homogeneous textures in a flexible and interactive way. Integrating artistic effects and performing a frames' animation is also possible, allowing the production of a variety of effects in an easy and general way.

Keywords

Texture synthesis, Non-homogeneous textures, Artistic effects, Image generation.

1. INTRODUCTION AND MOTIVATION

Textures reserve much attention and interest in Computer Graphics. Although in the last years many works concentrate in optimizing synthesis algorithms, the production of *ad hoc* generated outputs still remains challenging: recent research in this field recognizes the lack of local control in the synthesis process and there is a need for flexible user intervention [Ash01], [Her01]. The complex nature of real objects requires many attributes in order to be visualized in a realistic and credible way, textures often seems to be generated by a few underlying processes, and they are the result of several appearances that need to be appropriately combined together. For this purpose, we propose a novel idea, which introduces the required flexibility in textures generation. With respect to our previous work, we generalize here the texture synthesis method, proposing a much broader and complete methodology, still offering the user a simple and intuitive interface.

Posters proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

2. THE ALGORITHM

Our algorithm proceeds as follows. It is based on a *per-pixel* synthesis (refer to [Wei00] for details about the basics), but instead of just using an input sample, we facilitate of a set of inputs, using in addition a collection of filters and operators, which produce a heterogeneous matrix of samples in order to produce *ad hoc* generated output images. Moreover, in order to improve quality and time costs, we performed a multi-pass algorithm using *Image Pyramids* [Hee95].

3. SYSTEM

We built a comprehensive system that incorporates several settings and functionalities. This framework allows the user to flexibly design his output and to interact with the texture generation process: the user can select options, defining variable parameters to generate personalized textures. One or more samples (a *sample set*) may be used as source seed for the synthesis. Additionally, further operators we named *Vector Field, Filters* and *User controls* also may contribute in generating or modifying the output.

In the framework, the samples' set is constituted by an input *matrix* (Fig.1), whose dimensions belong to the range $[1, W*H]$, being W and H, respectively, the specified width and height of the output texture.

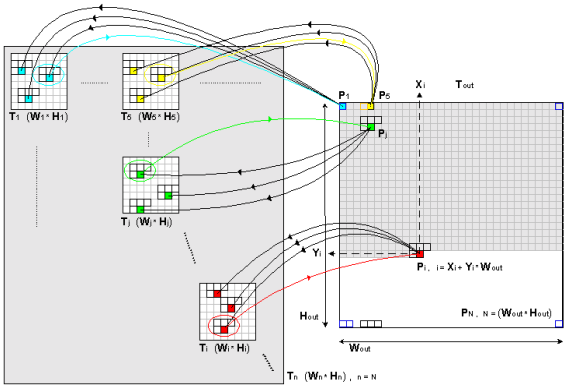


Figure 1. Matrix of input samples.

Some of the most common case studies and possible fields of application are:

- Filtered Texture Synthesis: *Image Processing Filters* may be used [Tap04] to vary characteristics in the sample or to modify and highlight some regions of interests in the output, as well as to augment a texture with artistic effects (Figure 2a).
- Texture Bending: superimposing a direction field may strongly influence a texture appearance,adding possible 3d effects and forcing the input patterns to follow specified directions (see Figure 2b).

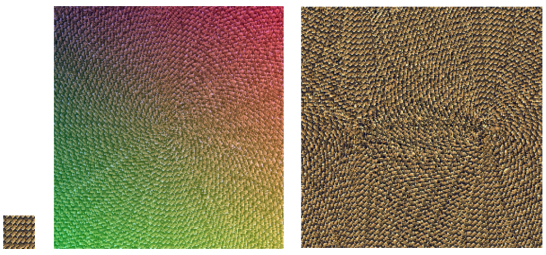


Figure 2. Gradually filtering the texture sample (a) and moving a given pattern along specified directions (b).

- Texture Synthesis for Vector Fields Visualization: using samples characterized by having a major direction (*directional textures*) leads to the interesting application of vector fields' visualization. Emphasizing ROIs, critical points (Figure 3) and singularities is possible (details about the procedure in [Tap03]).
- Texture Mixture and Texture Metamorphosis: objects often comprise more than one uniform look: they may present many appearances from different materials. Composing together dissimilar aspects is an important task for texturing natural 3d objects. Applying our method, this task can be achieved without the need of blending functions. Morphing, as well as changing textures aspect depending on surrounding circumstances or lighting effects is another interesting point to investigate in.
- Texture Animation: textures may evolve or change in time and space; a particular feature may appear, vary, move or disappear, similarly as for vector fields that describe complex physical phenomena.

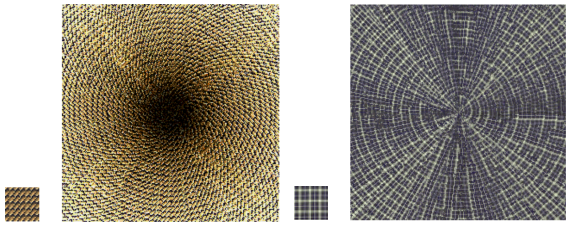


Figure 3. Enhancing center position with brightness (a) and just visualizing a given pattern along circular field.

4. CONCLUSIONS

We propose a system for flexible non-homogeneous texture synthesis. This novel approach aims at producing varying textures and patterns. With the described framework, we provide a complete tool for texture synthesis manipulation, which permits to generate a variety of outputs and effects, both for scientific and artistic tasks. Therefore, one of the main contribution of this work is having added various degrees of freedom to the generation of textures. We think this may be important in Computer Graphics as useful instrument, which contributes promising applications in several fields and also keeping the road open for further research. The extension to frames' animation still requires further investigation and it is part of our ongoing work.

5. REFERENCES

[Ash01] Michael Ashikhmin. Synthesizing natural textures. In *2001 ACM Symposium on Interactive 3D Graphics.*

[Hee95] David J. Heeger and James R. Bergen. Pyramid-based texture analysis/synthesis. *Proceedings of SIGGRAPH 9*5, pages 229–238.

[Her01] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. Image analogies. In *Proceedings of ACM SIGGRAPH 200*1, pp. 327–340, August 2001.

[Tap03] F. Taponecco and M. Alexa. Vector Field Visualization using Markov Random Field Texture Synthesis. EG/IEEE TCVG Visualization Symposium 2003, Grenoble, France.

[Tap04] F. Taponecco. User-defined texture synthesis. WSCG 2004, Plzen, Czech Republic.

[Wei00]L.Y.Wei and M.Levoy. Fast texture synthesis using tree-structured vector quantization. *Proceedings of SIGGRAPH 2000*, pp 479–488.

Robust Optical Measurement of Dust Thickness on a Flexible Filter Surface

Martina Uray
Graz University of Technology
Inffeldgasse 16
A–8010 Graz
Austria
uray@icg.tu-graz.ac.at

Matthias Rüther
Graz University of Technology
Inffeldgasse 16
A–8010 Graz
Austria
ruether@icg.tu-graz.ac.at

Horst Bischof
Graz University of Technology
Inffeldgasse 16
A–8010 Graz
Austria
bischof@icg.tu-graz.ac.at

ABSTRACT

A stereo computer vision system as a thickness measurement device is introduced. The challenging task is to measure the thickness of dust deposits over the surface of a flexible cloth filter with a target resolution of $0.5mm^2$ per measurement in the filter plane and a depth resolution of up to $50\mu m$. The total area to be measured is $100mm \times 2500mm$. There is practically no texture present on the dusty filter surface. Analyzing dust deposition over the whole filter length during operation has not been possible so far and promises to give insight to air flow in the filter chamber, behaviour of different filter materials and design flaws of filter chambers. Our approach is to first create dense 3D-reconstructions of surface patches. This is done using a calibrated stereo setup and a standard slide projector that provides texture by projecting a random pattern onto the filter surface. The thus obtained surface patches are registered against each other using an ICP algorithm to get a complete 3D model of the filter surface. This step is done twice, for the clean filter surface and for the surface including dust deposits. We finally retrieve the actual depth information by registering the clean and dusty models using markers applied to the filter surface and calculating the difference. In this paper we present the measurement device as well as the underlying vision system and show that it is possible to meet the given requirements under industrial conditions.

Keywords

Computer Vision, Stereo Reconstruction, Rigid Registration, ICP

1 INTRODUCTION - SYSTEM OVERVIEW

The given setup consists of several cylindrical filter bags (height = $2500mm$, diameter $\sim 120mm$) enclosed in a chamber with a glass window. Air mixed with chalk dust is blown into the chamber and sucked through the filter bags. The dust particles remain on the filter surface and form a layer of $\sim 3mm$ thickness, called filter cake.

Conference proceedings ISBN 80-903100-7-9
WSCG2005, January 31-February 4, 2005
Plzen, Czech Republic.

The goal is to measure cake thickness over the visible surface area of the filter bag, through the window at a distance of $\sim 500mm$. Reconstruction of patches of dense 3D models of the filter surface is done using a stereo vision setup. For alignment the popular Iterative Closest Points algorithm ([Besl92]) is applied.
Getting from the surface model to cake thickness one needs to reconstruct the filter surface in two states, clean and dirty. Since the surface is flexible and may change its position during operation, special Teflon markers are attached to the filter bag that remain clean during the filter process. Their exact position can be extracted in both states and are used as a landmark for another rigid registration step using a linear least squares approach.

2 RECONSTRUCTION

The camera setup is calibrated, so the two camera matrices P and P' are known, as well as the

fundamental matrix F and a set of homologous points, $\mathbf{u}_i, \mathbf{u}'_i$, from the calibration target.
So in our application the reconstruction of the 3D-points is composed of four steps: Projective rectification, matching, subpixel refinement and triangulation.

3 REGISTRATION

With the given hardware setup several small surface patches with an overlap of $\sim 5mm$ are reconstructed and registered against each other to form a continuous model. The registration of two of these patches is done using the ICP (iterative closest points) algorithm from [Besl92].
The final registration matrix is a combination of all matrices calculated in the iteration loops.
For fast results of the registration only the overlapping parts of the images are considered.

4 EXPERIMENTS

4.1 Reconstruction Results

For reconstruction accuracy the goals were to reach a resolution of 2 measurements per mm^2 over the filter surface. Until now tests were conducted using the laboratory setup only (i.e. without the looking glass in the optical path).
Since there was no highly accurate calibration target available, relative accuracy and depth resolution were evaluated by reconstructing two parallel planar surfaces with a height difference of $\sim 75\mu m$. The used target consists of a small plate with adhesive tape attached to the surface. The target was reconstructed in two different orientations and the average normal distance of the measured points from the best fit plane and the average height difference were evaluated. The interesting question was how much deviation from an ideal plane one would encounter in the planar regions and if the height of the tape could still be resolved. A height difference of about $75\mu m$ could not be measured exactly but was still clearly resolved. The resolution of $0.5mm^2$ in x/y direction could be easily reached in the average case, although larger areas may occur where no reliable matches can be found.

4.2 Registration Results

In our experiment the difference of two overlapping image parts is composed of a translation less than $1mm$ and a rotation less than $3°$.
The reconstructed 3D-models contain ~ 223000 points from which, for registration, 5000 points were regularly sampled. Several tests with Gaussian noise levels ($0mm \leq noise \leq 0.5mm$) were done. The rotation was fixed to an angle of $2°$ around the y-axis and for translation a vector of length $1.67mm$ with an arbitrary direction was used. At all tested noise levels the ICP algorithm achieved satisfactory results with an average Euclidean error less than $4mm$.
In addition we tested the ICP algorithm at a fixed noise level of $0.02mm$ and a fixed translation vector of length $1.67mm$. A stepwise rotation around the y-axis was calculated for different angles. A robust registration is possible for angles up to $10°$.
Finally we examined the behavior of the registration for a noise level of $0.02mm$ and a $2°$ rotation around the y-axis. The initial point set is translated along each coordinate-axis and the length of the translation vector is continuously increased. As long as the translation vector is less than $1mm$ (as given in our experiments) the average Euclidean error is less than $0.1mm$ afterwards the error increases noticeable.

5 CONCLUSION

This work shows that it is possible to conduct dense and accurate depth measurements using a stereo setup on practically untextured surfaces. The advantage over standard structured light approaches using coded light is that the target can be reconstructed in a single step and there is no need for expensive calibrated lighting equipment. The accuracy requirements can be reached under laboratory conditions. Ongoing work includes porting of the setup to the deployment stage and incorporating more sophisticated algorithms in the registration stage.

References

[Besl92] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions*, 14(2):239 – 256, 1992.

[Chen91] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. volume 3, pages 2724–2729, 1991.

[Gleas90] Shaun S. Gleason, Martin A. Hunt, and W. Bruce Jatko. Subpixel measurement of image features based on paraboloid surface fit. *Proc. Machine Vision Systems Integration in Industry, SPIE*, 1386, 1990.

[Rusin01] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the ICP algorithm. In *Proc. of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152, 2001.

Principal Component Analysis for the Approximation of an Image as an Ellipse

Sudanthi N.R. Wijewickrema
Computer Science & Software Engineering,
Monash University,
Clayton, VIC 3168, Australia
snw@csse.monash.edu.au

Andrew P. Papliński
Computer Science & Software Engineering,
Monash University,
Clayton, VIC 3168, Australia
app@csse.monash.edu.au

ABSTRACT

In this paper, we investigate a method of using principal component analysis(PCA) to fit an encapsulating ellipse to the image of a hypothetically ellipsoidal object. This technique is aimed at applications such as fruit sorting, where resource constraints and speed requirements necessitate the approximation of data.

Keywords

Ellipse Fitting, Principal Component Analysis (PCA)

1. INTRODUCTION

Principal component analysis (PCA) is a classical statistical technique which analyses the covariance structure of multivariate data [Hot33]. It determines the directions along which the variation of data occur and the corresponding importance of that direction. The first principal component gives the direction where the maximum variance could be observed. The second principal component is the direction of the next maximal variation and is orthogonal to the first and so on.

We take the image of an ellipsoidal object, represent it as a set of data points and use a PCA based algorithm to approximate it as an ellipse. Furthermore, we define a measure for calculating the error of fit, and outline a simple technique to determine the goodness of fit.

2. EXISTING WORK

Existing ellipse fitting algorithms could be categorized into Hough transform based methods and least squares fitting algorithms [HF98].

Hough transform based methods concentrate on mapping the data space to parameter space. Then the most likely parameters are chosen usually by clustering. They are robust against noise but have high computation time.

Least squares fitting looks at minimizing a distance measure between the set of points and the fitted curve. These methods are computationally better but are very sensitive to outliers.

Additionally, with respect to the target application, edge detection has to be done on the image to obtain the data points. In comparison, the method we propose only requires a simple segmentation of the image and needs only a few simple operations to calculate the ellipse parameters, making it ideal for resource constrained environments. A similar method is discussed in [Fan] where an ellipse is fitted to a region of interest.

WSCG POSTER proceedings *ISBN 80-903100-8-7*
WSCG'2005, January 31 - February 4, 2005,
Plzen, Czech Republic.

3. METHOD OF CALCULATION

The proposed method of ellipse fitting is as follows:

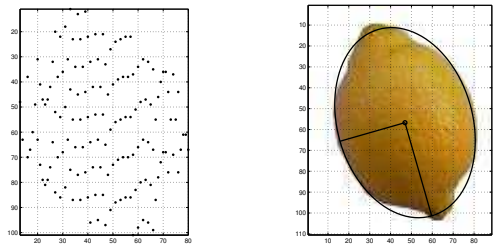
1. Acquiring the set of data points
2. Calculating the covariance matrix
3. Solving the eigensystem
4. Fitting the ellipse

3.1 Acquiring the Set of Data Points

Before performing PCA, a set of data points should be extracted from the image. To simplify operations, the image is converted into gray scale first.

It is assumed here that the image is acquired under controlled conditions and this is easily satisfied in the application that we aim for: i.e. fruit sorting. Next, global thresholding is used to segment the object from the background to extract the set of points. A more sophisticated segmentation method could be used for uncontrolled systems.

Decimation of the points resulting from the segmentation is done next, to reduce the processing. We go for computational simplicity, selecting every d^{th} pixel from the image that falls inside the object. If d is co-prime with R or C, (the no. of rows or columns of



(a) Decimated Image (b) Fitted Ellipse

Figure 1: An Example of a fitting

the image respectively, depending on the direction of selection: column-wise or row-wise), it should give good random uniform distribution. See figure 1(a).

3.2 Calculating the Covariance Matrix

After acquiring the data, the mean vector $\mathbf{c} = [c_1\ c_2]$ is calculated to get the center of the ellipse. Then the mean is removed to center the data around the origin. Let X consist of n 2-dimensional data points. The covariance matrix S, which indicates how strongly correlated the components are, is calculated as follows:

$$S = \frac{1}{n-1}\hat{X}^T \cdot \hat{X} \quad \text{where,} \quad \hat{X} = X - \mathbf{c} \qquad (1)$$

3.3 Solving the Eigensystem

We calculate the two eigenvalues, λ_i and eigenvectors V_i of S as follows:

$$\lambda_{1,2} = \frac{1}{2}\left(\sigma_{11} + \sigma_{22} \pm \sqrt{(\sigma_{11}-\sigma_{22})^2 + 4\sigma_{12}^2}\right) \qquad (2)$$

$$v_1 = \frac{\sigma_{12}}{\sqrt{(\lambda_1 - \sigma_{11})^2 + \sigma_{12}^2}}, \quad v_2 = \frac{\lambda_1 - \sigma_{11}}{\sqrt{(\lambda_1 - \sigma_{11})^2 + \sigma_{12}^2}} \qquad (3)$$

where, $V_1 = [v_1\ v_2]^T$ and $V_2 = [v_2\ -v_1]^T$

3.4 Fitting the Ellipse

The mean of data, c_1, c_2, the eigenvalues $\lambda_{1,2}$, and the direction of the first eigenvalue, ϕ, uniquely define the ellipse [Hot33]. Eqn (4) shows this and figure 1(b) shows an ellipse, fitted to an image of a fruit.

$$x_1 + jx_2 = 2(\cos t + j \sin t)e^{j\phi} + (c_1 + jc_2) \qquad (4)$$

4. ERROR & GOODNESS OF FIT

The error of fit determines how good a certain ellipse fitting algorithm is. For this, we define an error coefficient e, the normalized sum of squares of algebraic distances(D) from the edge points as shown in eqn (5).

$$e = \frac{1}{n}\sum_{i=1}^{n} D_i^2 \quad \text{where,} \quad n = \text{no. of edge points} \qquad (5)$$

The error of fit changes with the decimation constant d. To generalize the selection of d, we define a value G, as shown in eqn (6), so that an optimal set of points that minimizes the processing and produces an acceptable error of fit can be selected. See figure 2.

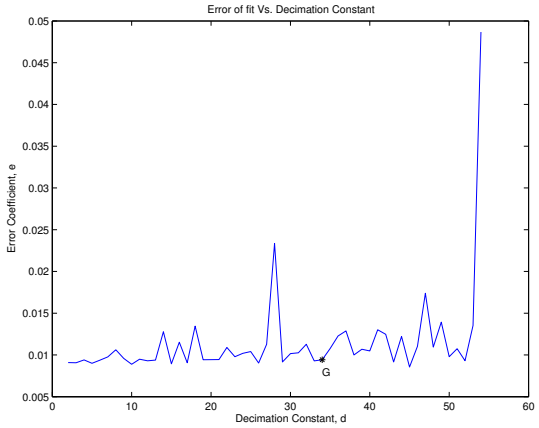


Figure 2: Error of fit Vs. Decimation constant

$$G = \text{round}(R \times \frac{3-\sqrt{5}}{2}) \qquad (6)$$

The goodness of the ellipse fit can then be calculated to test the feasibility of the approximation. For this we propose a simple method to suit the target application. We define a threshold T, and take any fit that has an error measure below it as acceptable. The threshold is determined experimentally to suit the application.

5. CONCLUSION

The application of PCA to estimate the image of an object as an ellipse was investigated as a technique to be used in fruit sorting applications.

The processing here has to be done in real-time, and hence the algorithm and techniques used are aimed at simplifying the calculations and maximizing efficiency. For instance the use of a very simple segmention method was used to aquire the data points. This does not compromise the accuracy as the images are obtained under controlled conditions. Decimation of data was also done to achieve higher speed.

The above discussed method of representating an image of an object as an ellipse could therefore be considered as an efficient and suitable method to be used in fruit sorting in real-time and could be easily adapted for other similar applications.

6. REFERENCES

[Fan] Fit an Ellipse to a Region of Interest. www.dfanning. com/ip_tips/fit_ellipse.html.

[HF98] R. Halír and J. Flusser. Numerically stable direct least squares fitting of ellipse. In *Proceedings of the 6th Conference in Central Europe on Computer Graphics and Visualization*, pages 253–257, Plzen, February 1998.

[Hot33] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24, 1933.

Generalized Hebbian Learning for Ellipse Fitting

Sudanthi N.R. Wijewickrema
Computer Science & Software Engineering,
Monash University,
Clayton, VIC 3168, Australia
snw@csse.monash.edu.au

Andrew P. Papliński
Computer Science & Software Engineering,
Monash University,
Clayton, VIC 3168, Australia
app@csse.monash.edu.au

ABSTRACT

In this paper, we investigate the use of a neural network employing Genralised Hebbian Learning for the approximation of an image of a hypothetically ellipsoidal object as an ellipse. Further, we discuss how the same algorithm is used with higher dimensional data to model hyperellipsoids, with the basic aim at a specific application, namely the modelling of an object as an ellipsoid given a set of 3-dimensional points.

Keywords

Ellipse Fitting, Principal Component Analysis (PCA), Generalised Hebbian Learning(GHL)

1. INTRODUCTION

An algorithm for fitting an ellipse to the image of an ellipsoidal object using principal components was discussed in [WP04]. It suffers from the problem of not being easy to extend to multiple dimensions.
We discuss how Generalised Hebbian Learning, which is an Artificial Neural Network (ANN) approach of performing PCA on a set of data, is used in association with this algorithm to estimate the ellipse parameters in 2-dimensions. We further investigate the extension of this algorithm to multiple dimensions, with the major objective of 3-dimensional ellipsoid modelling.

2. EXISTING WORK

Most existing algorithms for ellipse fitting either fall into Hough transform based methods or least squares fitting algorithms [HF98]. In contrast, [WP04] discusses a method based on PCA. It was found that this method has an advantage over other methods with respect to speed and robustness against outliers. This makes it ideal for resource constrained environments.
This paper looks at extending the above mentioned algorithm so that it could be used for hyperellipsoid fitting in multiple dimensions.

WSCG POSTER proceedings *ISBN 80-903100-8-7*
WSCG'2005, January 31 - February 4, 2005,
Plzen, Czech Republic.

3. GHL FOR ELLIPSE FITTING

It was shown in [Hay94] that a network employing the learning rule known as Generalised Hebbian Learning (GHL) performs PCA for zero mean inputs. We use this principle to calculate the principal components and the variance of data along them. Then using them we determine the ellipse parameters [WP04]. The algorithm for using GHL for ellipse fitting is as follows:

- Arrange the points in a $m \times N$ matrix, x
  where, N - No. of points
  m - No. of dimensions/neurons($m = 2$ for 2-D)
- Initialize the $m \times m$ weight matrix, w with random values
- Calculate the sum of the initial length of the weight vectors
  $l = \sum w_j^2$
- Define the length convergence error, e
- Define the learning rate, η
- While the absolute value of the length of the weight vectors have not converged to 1
  - Calculate the output
    $y = W * x$
  - Calculate the weight update
    $\Delta W = \eta y(x^T - y^T W)$
  - Update the weight matrix with the weight updates
    $W = W + \Delta W$
  - Update the length of the weights
    $l = \sum w_j^2$

- Determine the eigenvectors of the input correlation matrix by the weight matrix
  $V = W^T \quad \Rightarrow \quad v_i = [V_{1,i} \cdots V_{m,i}]^T$
- Calculate the eigenvalues by the variance of the output values
  $\lambda = var(W.x)$
- Determine the axes of the ellipse using the eigenvalues and its orientation using the eigenvectors or principal components
  $x_1 + jx_2 = 2(\cos t + j \sin t)e^{j\phi} + (c_1 + jc_2)$
  where c_1, c_2 are the mean of the data,
  and $\tan \phi = v_2/v_1$
- Draw the ellipse around the points or the image

Figure 1 shows an example of an ellipse fit. In the case of an image, the method outlined in [WP04] should be used to sample the points before fitting the ellipse.

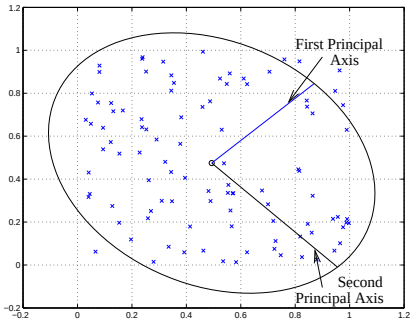


Figure 1: An example of ellipse fitting

4. ELLIPSOID MODELLING

The extension of the above algorithm to fit hyper-ellipoids in higher dimensions is done by changing the number of neurons of the neural network. This changes the input, weight and output matrices accordingly. The parameters are then determined by using the eigenvectors and eigenvalues as in the 2-D case. Figure 2 shows an example in 3-dimensions.

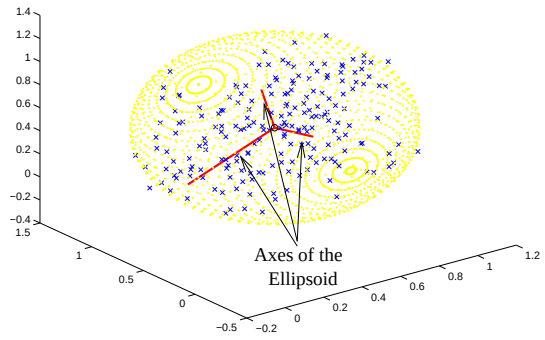


Figure 2: An example of 3-D ellipsoid modelling

5. COMPARISON OF METHODS

The neural network method discussed in this paper to calculate the ellipse parameters is compared with that used in [WP04]. The main advantage here is the simple extensibility to higher dimensions.
Further this method gives the user control over the error of fit. By changing the length convergence error, the processing and accuracy could be balanced to suit the requirements of the application.
It was also found that the above discussed method is faster, specially in higher dimensions. Figure 3 shows this comparison of average processing times, where **PCA** is the algorithm discussed in [WP04] and **GHL** is the method discussed above.

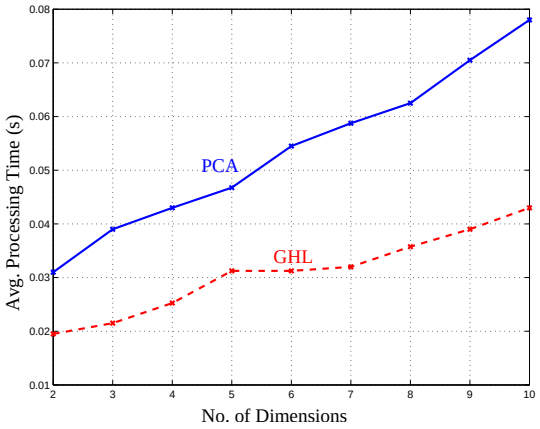


Figure 3: Processing time Vs No. of dimensions

6. CONCLUSION

In this paper we discussed how a neural network employing Generalized Hebbian Learning could be used to fit an ellipse to a set of 2-dimensional data points and how to extend this algorithm to n-dimensions.
The main strength of this algorithm lies in its extensibility to higher dimensions without substantially increasing the amount of processing.
Hence we conclude that this method is acceptable in ellipse fitting but is more efficient in higher dimensions. It could be used in applications where resource constraints require the approximation of data to minimize the processing and maximize speed.

7. REFERENCES

[Hay94] S. Haykin. *Neural Networks – a Comprehensive Foundation*. Macmillan, New York, 1994. ISBN 0-02-352761-7.

[HF98] R. Halír and J. Flusser. Numerically stable direct least squares fitting of ellipse. In *Proceedings of the 6th Conference in Central Europe on Computer Graphics and Visualization*, pages 253–257, Plzen, February 1998.

[WP04] S. N. R. Wijewickrema and A. P. Papliński. Principal component analysis for the approximation of an image as an ellipse. Technical Report 2004/160, Computer Science and Software Engineering, Monash University, Australia, September 2004.

A Study on a Cyber World for Language Acquisition and Sensory Information Transfer Control

Yuya KATAYAMA Minoru OKADA
Graduate School of Information, Production and Systems, Waseda University
7 Hibikino-2, Wakamatsu-ku, Kitakyushu 808-0135 Japan
u8ktym@suou.waseda.jp mokada@waseda.jp

ABSTRACT

In this paper sensory information transfer methods for sight, sound, and text chat are proposed. The authors have studied a foreign language acquisition system *Orbis* utilizing cyber worlds. For such a system, realistic communication, and the server CPU load and the network traffic become important issues. A concept of *communication space* to control sensory information in a cyber world is introduced and discussed.

Keywords

The Internet, Multi-user, VRML, Virtual Environment, Avatar, Cyber World

1 INTRODUCTION

The authors proposed a foreign language acquisition system *Orbis*[CBoudreau98]. It is intended to provide places and chances to communicate with native speakers in a shared cyber world on the web. Then it becomes important to construct the place in which users can experience realistic communication and unnecessary sensory information is eliminated. In this paper we propose a concept of *communication space* which controls sensory information in a virtual environment.

2 ORBIS ARCHITECTURES

In the Orbis, a client-server model synchronization of avatar information enables users to share a virtual environment. Autonomous characters, behaving as an avatar without the actual user, is recognized as a client at the Orbis server; implemented with the same protocol as an ordinary client. Prototype of autonomous characters only do autonomous walk and tiny conversation. The text chat function is implemented with a Java frame component (Figure 1), and synchronizes the chat messages with a client-server model.

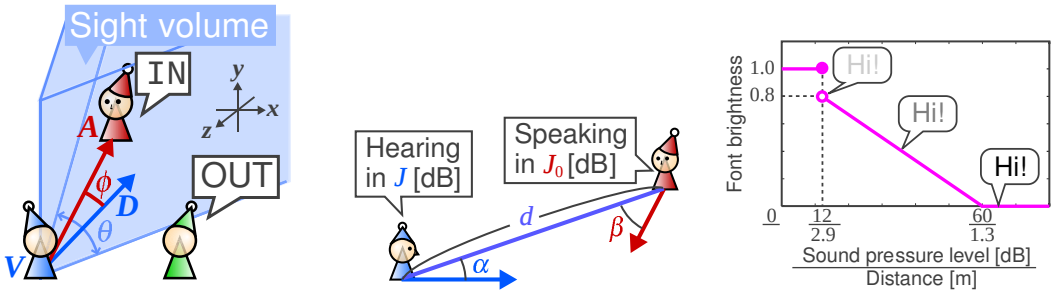
POSTERS proceedings ISBN 80-903100-8-7
WSCG '2005, January 31-February 4, 2005
Plzen, Czech Republic.

Figure 1: Main and chat frames

3 COMMUNICATION SPACE

3.1 Sight Communication Space

For the system many people logged in, the server CPU load dominated with the amount of dispatch processes becomes problematic. As a solution, there is a method that the server sends a scene by each divided region instead of avatar information[DMinoura98], however; the rendering reality decrease. Therefore, we propose a concept of a sight communication space which restricts the dispatching to the avatars in his sight. *E.g.* the case of Figure 2 (a), the client of the green avatar is not required to receive the blue avatar's information. We performed experiments to evaluate the availability of the sight communication space by measuring the dispatch ability of the Orbis server using autonomous characters keep updating themselves. In the experiments, the total client's update counts α_C and the server's dispatch counts α_S were measured. In Figure 3, $(n-1)\alpha_C$ and α_S are plotted with lines marked ■ and ●, respectively. The horizontal solid line indicates the bandwidth limit caused by the network. The measurement value missed from around $n = 10$, and saturated around 45,000 [times/sec]. So this means



(a) Sight (b) Sound (c) Text chat

Figure 2: Three types of communication spaces

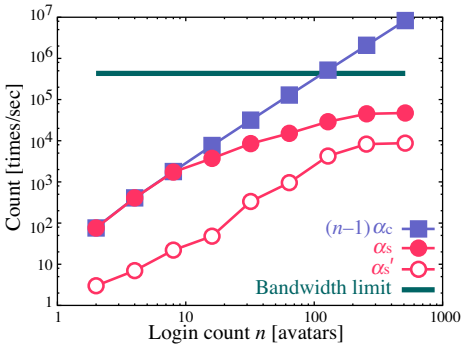


Figure 3: Dispatch ability of the server

the limit of CPU processing at the server. As the result with the sight communication space, the dispatch count α'_s, plotted with lines marked ○ in Figure 3, was reduced to approximately 1/4 at least, and the CPU load was reduced to approximately 1/4, similarly.

3.2 Sound Communication Space

In a virtual environment, controlling the loudness of voice according to the distance d between a speaker and a listener can improve the immersion[HNakanishi99]. Referring the speaker's sound pressure level J_0[dB], the angles α, β and the distance d[m] are prepared as illustrated in Figure 2 (b). The sound communication space defines the sound pressure level J[dB] at the listener by the following function.

$$J = \frac{J_0 f(\alpha) g(\beta)}{d^2} \qquad (1)$$

Where the functions of angles, $f(\alpha)$ and $g(\beta)$, are in a range $[0, 1]$, and cosine is actually used for them.

3.3 Text Chat Communication Space

Chatting with texts in a large virtual space, messages from avatars stay so far might be excess information; the text chat commutation space is used to eliminate the message from a position which cannot be seen by the sight communication space. Furthermore the font brightness of the message displayed at the chat window is changed by the same function described in 3.2. According to eq. (1), where J_0 is 100[dB], the font brightness B is determined by a function as shown in Figure 2 (c). If B becomes 1 the message is not displayed at the chat frame, because when $B > 0.8$, the font is almost invisible. In the case that a user wants to call to avatars at the place the message is not sent, the Loud function makes the message be dispatched to the all avatars in less than 10[m] distance without above conditions.

There are found some problems such as, (1) closeness of conversation range, (2) difficulty to know where avatars chatting up, and (3) difficulty to know whether sent message was reached. For (1), the transfer functions of sound and text chat communication are problematic. In eq. (1), cosine functions for $f(\alpha), g(\beta)$ and the denominator d^2 are needed to improve. For (2), a simple overhead view map can be helpful to communicate easily. Then for (3), an implementation of a new function to announce users whether an input message was sent might be required.

4 CONCLUSION

This paper has described a concept of communication space to improve the cyber world environment. The three-types communication space, sight, sound and text chat, corresponding to main sensory information of the Orbis, are introduced. One of the future works is to set and to improve the transfer function of sensory information for each communication space.

References

[CBoudreau98] Boudreau, C. *et al.*Foreign Language Exposure in Shared Virtual Environment. Proc. of VSMM'98, pp.54–59, 1998.

[DMinoura98] Minoura, D. *et al.*A Method to Display a Large Number of Participants in Multi-User 3D Cyberspace. IEICE Trans. Inf. & Syst., Vol.J81-D2, No.5, pp.962–971, 1998.

[HNakanishi99] Nakanishi, H. *et al.*FreeWalk: A 3D Virtual Space for Casual Meetings. IEEE Multimedia, Vol.6, No.2, pp.20–28, 1999.

Anatomy-based Human Face Reconstruction Using Multi-layer Deformation

Yu Zhang
School of Computing
National University of Singapore
117543, Singapore
zhangy@comp.nus.edu.sg

Terence Sim
School of Computing
National University of Singapore
117543, Singapore
tsim@comp.nus.edu.sg

Eric Sung
School of Electrical and Electronic Engineering
Nanyang Technological University
639798, Singapore
eericsung@ntu.edu.sg

ABSTRACT

We present a method for reconstructing anatomy-based facial models with minimal manual intervention. The technique is based on deforming a multi-layered prototype model to the acquired surface data in an "outside-in" manner. It enables a variety of applications, including texture transfer, morphing, statistical analysis of the face shape, and face recognition database satisfying large extra- and intra-subject variations.

Keywords

face reconstruction, facial animation, anatomy-based model, deformation.

1. INTRODUCTION

Current dense surface measurement techniques allow us to generate precise 3D shapes of faces by using 3D shape acquisition systems such as a range scanner, a stereo photogrammetry system, or an active light striper. However, using the range data for face reconstruction suffers from several key problems: 1) absence of animation structure; 2) irregular and dense surface data; and 3) incomplete data. We propose a method for efficiently reconstructing animatable 3D faces from range data. It is based on deforming a prototype model to the acquired surface data in an "outside-in" manner: deformation applied to the external skin layer is propagated, with the final effect of deforming the underlying muscles.

2. PREVIOUS WORK

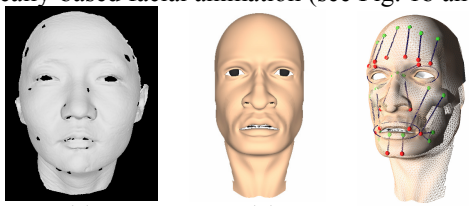
Previous major researches on 3D face reconstruction from range data include studies on conforming a generic face mesh to the located facial features in a 2D depth image [Wat91, Lee95], a study on deforming a subdivision surface for fitting [Mar00], and a study on morphing a generic model with scattered data interpolation [Kah02]. Our technique differs from these methods in that 1) it uses a dynamic deformable model for the skin layer deformation in 3D space; 2) it automatically adapts the underlying muscle layer that includes three types of muscles; and 3) it is capable of generating animatable models from incomplete scanned data.

Conference proceedings ISBN 80-903100-7-9
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

3. MULTI-LAYER DEFORMATION

Data Acquisition

The range data acquired by a Minolta VIVID 900 Digitizer is a dense point surface, frequently with holes due to missing data (see Fig. 1a). We have developed an anatomy-based prototype model that incorporates a skin-muscle-skull structure for physically-based facial animation (see Fig. 1b and c).



(a) (b) (c)

Figure 1. (a) Scanned data. (b), (c) Prototype

Skin Layer Deformation

To accomplish the fitting, a physical model based on the mass-spring-damper (MSD) system is created from the skin mesh of the prototype model: each mesh vertex is characterized by a mass m and each mass point is linked to its neighbors by damped

massless springs. The discrete Lagrangian equations of motion for the MSD mesh can be expressed in 3D vector form as:

$$\mathrm{M}\frac{d^2x}{dt^2}+\mathrm{D}\frac{dx}{dt}+\mathrm{F}_e(x,\mathrm{K})=\mathrm{F}_d(x)+\mathrm{F}_f(x) \qquad (1)$$

Given N vertices, x represents a $3N$ vector of nodal displacement, M, D, and K are $3N\times3N$ matrices describing the mass, damping, and stiffness between vertices, respectively. F_e, F_d and F_f are $3N$ vectors, representing the elastic force, data force and feature force, respectively.

A vertex x_i receives the data force from the closest scanned data point p_j. Since the scanned data is used to attract the generic mesh and deform it into a consistent shape, we use a type of springs with natural length of zero for this kind of deformation:

$$F_e(x_i)=\alpha\frac{\|x_i-p_j\|}{H}\overrightarrow{x_ip_j} \qquad (2)$$

where α controls the force strength, and H is a normalizing constant.

We have interactively specified a set of 23 landmarks on both the prototype model and the scanned data to identify facial features (e.g. eye and mouth corners, nose tip etc). For each landmark s_k on the scanned surface, it has a counterpart q_k on the generic mesh. The force generated by s_k is applied to q_k and its neighboring vertices:

$$F_f(x_i)=\begin{cases}\beta\dfrac{\|x_i-s_k\|}{L}\overrightarrow{x_is_k} & \text{if } d_{ik}<Q_k^{ref}\\ 0 & \text{otherwise}\end{cases} \qquad (3)$$

where β is a scaling parameter, L is a normalizing constant, d_{ik} denotes the topological distance of the vertex x_i from the landmark q_k measured in terms of the smallest number of edges between them, and Q_k^{ref} denotes the reference topological distance of kth landmark point. We integrate Eq. 1 using the explicit Verlet-Leapfrog method.

Muscle Layer Deformation

The muscles incorporated in the prototype model are divided into three types: the *linear muscle*s, *sheet muscle*s, and *sphincter muscles*. The central muscle fiber of the first two types of muscles has one end attached to the skull which is called the *muscle attachment point* and the other end inserted to the skin which is called the *muscle insertion poin*t. For a muscle insertion point, it is located on the surface of an indexed skin mesh triangle. After deformation of the skin layer, its new position is easily found with this transformed triangle. For a muscle attachment point, we first find its counterpart on the skin along the direction of skull surface normal. They are related by a *tissue depth vector*. After deformation of the skin layer, the new position of the muscle attachment point is obtained by offsetting the transformed skin counterpart along the negated tissue depth vector.

The sphincter muscles consist of fibers that loop around the eyes and mouth and can draw towards a virtual center. They are modeled as a parametric ellipse. To adapt a sphincter muscle, we use one end of its two axes as reference points. By finding their skin counterparts on the undeformed skin surface, we obtain their new positions in the same way as to determine that of the muscle attachment point. The epicenter and half-lengths of two object axes of the elliptic muscle model are then calculated based on the new positions of the reference points.

4. RESULT

Fig. 2 shows the reconstructed facial model with Gouraud shading, its texture-mapped appearance, and adapted underlying muscle layer.

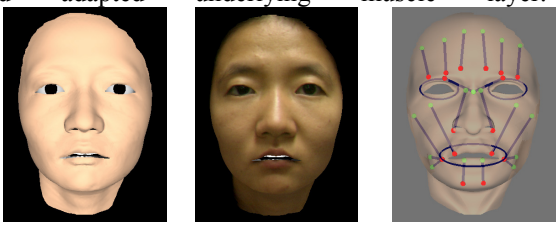


Figure 2. Reconstructed personalized face.

Our method creates a consistent surface parameterization of range scans, which gives immediate point correspondences between all the models via a single, prototype layout. This in turn enables a series of applications that involve multiple models simultaneously, including texture transfer, morphing, statistical analysis of the face shape space, and face recognition database satisfying a large extra-and intra-subject variations.

5. REFERENCES

[Kah02] K. Kahler, J. Haber, H. Yamauchi, and H. P. Seidel. Head shop: Generating animated head models with anatomical structure. in ACM SCA'02 conf.proc., pp.55-64, 2002.

[Lee95] Y. Lee, D. Terzopoulis, and K. Waters. Realistic modeling for facial animation. in ACM SIGGRAPH'95 conf.proc., pp.55-62, 1995.

[Mar00] S. Marschner, B. Guenter, and S. Raghupathy, Modeling and rendering for realistic facial animation. in EG Workshop on Rendering'00 conf.proc., pp.231-242, 2000

[Wat91] K. Waters and D. Terzopoulus. Modeling and animating faces using scanned data. Journal of Visualization and Computer Animation, vol.2, pp. 123-128, 1991.

Classification Techniques in Pattern Recognition

Lihong Zheng and Xiangjian He
Faculty of IT, University of Technology, Sydney
PO Box 123, Broadway NSW 2007, Sydney, Australia
{Lzheng, sean}@it.uts.edu.au

ABSTRACT

In this paper, we review some pattern recognition schemes published in recent years. After giving the general processing steps of pattern recognition, we discuss several methods used for steps of pattern recognition such as Principal Component Analysis (PCA) in feature extraction, Support Vector Machines (SVM) in classification, and so forth. Different kinds of merits are presented and their applications on pattern precognition are given. The objective of this paper is to summarize and compare some of the methods for pattern recognition, and future research issues which need to be resolved and investigated further are given along with the new trends and ideas.

Keywords

Pattern recognition, feature extraction, feature selection, mapping, kernels, support vector machines

1. INTRODUCTION

Pattern recognition can also be seen as a classification process. Its ultimate goal is to optimally extract patterns based on certain conditions and is to separate one class from the others. Pattern recognition was often achieved using linear and quadratic discriminants [1], the k-nearest neighbor classifier [2] or the Parzen density estimator [3], template matching [4] and Neural Networks [5]. These methods are basically statistic. The problem of using these recognition methods is having to construct a classification rule without having any idea of the distribution of the measurements in different groups. Support Vector Machine (SVM) [6] SVMs have gained prominence in the field of pattern classification. They are forcefully competing with other techniques such as template matching and Neural Networks for pattern recognition.

This paper is organized as follows. We first introduce some general process of pattern recognition and basic techniques in section 2. Conclusions are made in section 3.

2. GENERAL PROCESS OF PR

A pattern is a pair comprising an observation and a meaning. Pattern recognition is inferring meaning from observation. Designing a pattern recognition system is establishing a mapping from measurement space into the space of potential meanings, whereby the different meanings are represented in this space as discrete target points. The basic components in pattern recognition are preprocessing, feature extraction and selection, classifier design and optimization.

2.1 Preprocessing

The role of preprocessing is to segment the interesting pattern from the background. Generally, noise filtering, smoothing and normaliztion should be done in this step. The preprocessing also defines a compact representation of the pattern.

2.2 Feature Selection and extraction

Features should be easily computed, robust, insensitive to various distortions and variations in the images, and rotationally invariant. Two kinds of features are used in pattern recognition problems. One kind of features has clear physical meaning, such as geometric or structural and statistical features. Another kind of features has no physical meaning. We call these features mapping features. The advantage of physical features is that they need not deal with irrelevant features. The advantage of the mapping features is that they make classification easier because clear boundaries will be obtained between classes but increasing the computational complexity.

Feature selection is to select the best subset from the input space. Its ultimate goal is to select the optimal features subset that can achieve the highest accuracy results. While feature extraction is applied in the situation when no physical features can be obtained.
Most of feature selection algorithms involve a combinatorial search through the whole space. Usually, heuristic methods, such as hill climbing, have to be adopted, because the size of input space is exponential in the number of features. Other methods

Conference proceedings ISBN 80-903100-8-7
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

divide the feature space into several subspaces which can be searched easily.

There are basically two types of feature selection methods: filter and wrapper [7]. Filters methods select the best features according to some prior knowledge without thinking about the bias of further induction algorithm. So these methods performed independently of the classification algorithm or its error criteria.

In feature extraction, most methods are supervised. These approaches need some prior knowledge and labeled training samples. There are two kinds of supervised methods used: Linear feature extraction and nonlinear feature extraction. Linear feature extraction techniques includle Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), projection pursuit, and Independent Component Analysis (ICA). Nonlinear feature extraction methods include kernel PCA, PCA network, nonlinear PCA, nonlinear auto-associative network, Multi-Dimensional Scaling (MDS) and Self-Organizing Map (SOM), and so forth.

2.3 Classifiers design

After optimal feature subset is selected a classifier can be designed using various approaches. Roughly speaking, there are three different approaches [14]. The first approach is the simplest and the most intuitive approach which is based on the concept of similarity. Template matching is an example. The second one is a probabilistic approach. It includes methods based on Bayes decision rule, the maximum likelihood or density estimator. Three well-known methods are K-nearnest neighbour (KNN), Parzen window classifier and branch-and bound methods (BnB).The third approach is to construct decision boundaries directly by optimizing certain error criterion. Examples are fisher's linear discriminant, multilayer perceptrons, decision tree and support vector machine. The important advantage of SVM is that it offers a possibility to train generalizable, nonlinear classifiers in high dimensional spaces using a small training set. SVMs generalization error is not related to the input dimensionality of the problem but to the margin with which it separates the data. That is why SVMs can have good performance even with a large number of inputs. There are many kinds of methods aiming at reducing the computational burden for pattern recognition. Examples are K-nearest neighbor method, Parzen Window, Clustering, PNN and Branch-and-bound. KNN' significant disadvantage is that the distance must be calculated between an unknown and every prototype each time a sample is recognized. Parzen Window depends on the kernel function and on the value of the window-width h. It allows us to obtain complex nonlinear decision boundaries. Clustering method aims at partitioning a given set of N data into M groups so that similar vectors are grouped together. PNN's main idea of the PNN can be generalized so that we can optimize multi-merging steps. BnB technique uses a search tree for finding the optimal clustering and generates clustering through a sequence of merging operations.

2.4 Optimization

The optimization is not a separate step, it is combined with several parts of the pattern recognition process. In preprocessing, optimization guarantee that the input pattern have the best quality. Then in the feature selection and extraction part, optimal feature subsets are obtained under some optimization techniques. Furthermore, the final classification error rate is lowered in the classification part.

3. CONCLUSION

The basic idea we get is: the more relevant patterns at your process, the better features subsets you obtain, the more simple your classifier will be applied, finally the better your decisions will be. Based on our analysis of various methods, a combination of various techniques may be a better way for our final goal that will utilize available domain knowledge to make decisions automatically and accurately. . In summary, we should attempt to design a hybrid system combining with multiple models.

4. REFERENCES

[1] R.A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems,"Annals of Eugenics, vol. 7, part II, pp. 179-188, 1936.

[2] Dasarathy, B.V.; "Minimal consistent set (MCS) identification for optimal nearest neighbor decision systems design", IEEE Transactions on Systems, Man and Cybernetics, Vol. 24, Issue: 3, pp:511 – 517, March 1994.

[3] Girolami, M.; Chao He; "Probability density estimation from optimally condensed data samples" Pattern Analysis and Machine Intelligence, IEEE Transactions on, Volume: 25, Issue: 10 , pp:1253 – 1264,Oct. 2003.

[4] Meijer, B.R.; "Rules and algorithms for the design of templates for template matching", Pattern Recognition, 1992. Vol.1. Conference A: Computer Vision and Applications, 11th IAPR International Conference on, pp: 760 – 763, Aug. 1992.

[5] Hush, D.R.; Horne, B.G.; "Progress in supervised neural networks", Signal Processing Magazine, IEEE, Vol. 10, Issue: 1, pp:8 – 39, Jan. 1993

[6] Vapnik, V., The Nature of Statistical Learning Theory, Springer, 1995.

[7] Julia Neumann, Christoph Schnorr, "SVM-based feature selection by direct objective minimization", 2004.

Segmentation of complex shapes by adaptive energy forces

Huiyu Zhou[1], Tangwei Liu[2], Faquan Lin[2], Yusheng Pang[2], Ji Wu[2]
[1]University of Essex, Colchester, United Kingdom, CO4 3SQ
[2]Guangxi Medical University, Guangxi, P.R. China, 530027

ABSTRACT

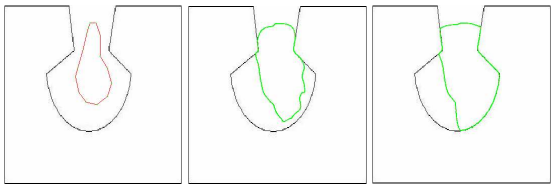
The classical gradient vector flow technique, to some extent has the ability to catch up dynamic topological changes, and therefore to extract complex shapes. Due to the reliance on the detected edges and the corresponding strength, the snake may be obstructed to rest on the ideal contours. To remedy these two deficiencies, a new deformable model is proposed in this paper. The idea is to improve the energy function by consistently reducing the Euclidean distance between the initial centroid and the estimated one of the snake. This is achieved by applying the mean shift for estimating the varied centroid of the snake during the iteration, which indicates the balance point of the overall forces. Experimental results show favorable performance of the proposed approach.

Keywords
Deformable model, gradient vector flow, topology, energy force, mean shift

1 INTRODUCTION

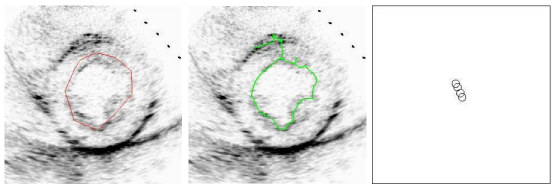
In this paper, we propose an improved gradient vector flow for segmenting concave regions. It has been observed that the classical snakes as well as the GVF cannot move towards the true concave boundaries *if the net effect of the internal and external (and damping) forces has reached zero during iteration*. An intuitive idea is to intelligently break this "false balance", and then encourage the snake to march until it is located at the real boundaries. Therefore, the segmentation problem anticipates an in-depth study on the energy dynamics, based on the deformable model as follows: $E_t = \int_\Omega(\alpha \parallel C(s)' \parallel^2 + \beta \parallel C(s)'' \parallel^2 + P(C(s)))ds$, where the former two terms represent the internal energy, and the later one is the external energy. Additionally, α and β are the tension and rigidity terms respectively. $C(s)$ is the contour that delineates the desired boundaries, and $s \in$[0,1]. P is defined as $P = - \parallel \bigtriangledown I \parallel$, where I is the image intensity.
Our research is motivated by seeking a new dynamic energy fomulation, which may be used to adapt the internal and external forces in case a false energy balance happens near to the real boundaries. To achieve this, a mixed model of the GVF scheme and the mean shift technique is hence proposed. This model is based on the fact that the centre of the weight of the snake normally is away from that of the ideal boundary when the former fails to settle on the latter. One example is illustrated in Fig. 2.

WSCG POSTERS proceedings ISBN 80-903100-8-7
WSCG'2005, January 31 - February 4, 2005
Plzen, Czech Republic.



(a) (b) (c)

Figure 1: An example that shows the performence of the GVF on a synthetic concave shape.



(a) Start (b) End (c) Centroid

Figure 2: Movement of the centroid of the snake during the iteration.

2 GVF snake

The GVF snake yields an external force field called *GVF field* in the continuous gradient domain [XP97]. Technically, a binary edge map is demanding, which forces the snake to effectively approach the edge-like areas. When the GVF snake is finally settled, where the internal and external forces are balanced, we shall have the relationship as

$$\alpha C''(s) - \beta C''''(s) + \gamma V = 0, \qquad (1)$$

where γ is a proportional coefficient. Alternatively, one can modify Eq. 1 to be

$$\lambda_\alpha C''(s) - \lambda_\beta C''''(s) + \gamma V = 0, \qquad (2)$$

where λ_α and λ_β stand for the contributions of the elasticity and rigidity in the internal energy term. To generate the representation for λ_α and λ_β, we introduce a mean shift based framework.

3 Mean shift

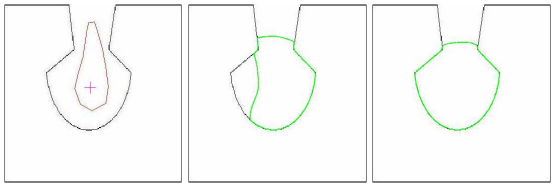
Mean shift is employed to search for a contour candidate that has the most similar characteristics to that of the target contour. To efficiently constrain the contour propogation, the CAMSHIFT algorithm by [Bra98] is used due to its preference of accounting for dynamically changing distributions during the evolution.

To exploit the CAMSHIFT algorithm, we first calculate the zeroth moment M_{00}, moment M_{10} for x-coordinates, and moment M_{01} for y-coordinates of image points on the contour candidate. This requires an estimate of the Euclidean distance between the origin (0,0) and individual points before the moment calculation is conducted. The centroid (x_c, y_c) of the contour is then calculated by $x_c = M_{10}/M_{00}$ and $y_c = M_{01}/M_{00}$. The Euclidean distance d_e between the initial centroid and the estimated one is consistently calculated so that λ_α and λ_β can be estimated by $\lambda_\alpha = \alpha d_e$, and $\lambda_\beta = \frac{\beta}{d_e}$. If $d_e \leq 1$ and Eq. 2 holds, then the evolution will stop as the convergence to the target contour has been reached; otherwise, the search has to continue.

4 EXPERIMENTAL RESULTS

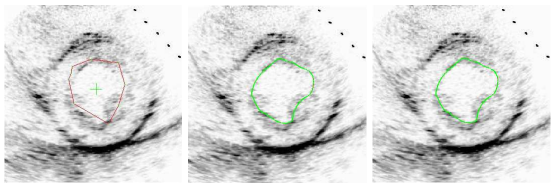
A couple of synthetic and real images have been tested using the proposed mean shift based algorithm. The first one is the synthetic image demonstrated in Fig. 1, where the GVF snake is leaking. Before the segmentation is conducted, the snake and its centroid need to be initiated, both of which have been illustrated in Fig. 3 (a). Fig. 3 shows that the proposed snake model properly outlines the concave shape.

The second one is a short-axis cardiac ultrasonic image. This human heart image consists of a number of



(a) (b) (c)

Figure 3: Segmentation of the synthetic image shown in Fig. 1, where '+' shows the position of the initial centroid ($\alpha = 0.05; \beta = 0; \gamma = 1; \mu = 0.1$).



(a) (b) (c)

Figure 4: Segmentation of the cardiac image shown in Fig. 2 ($\alpha = 0.05; \beta = 0; \gamma = 1; \mu = 0.1$).

structures besides the endocardial border shown in Fig. 4. According to Fig. 2 and 4, our approach has better performance on the border segmentation than the classical GVF algorithm.

5 CONCLUSION

This paper has presented a novel algorithm for image segmentation, integrating the classical GVF algorithm and the mean shift technique. The experimental results demonstrate that the new approach has favourable performance in segmenting concave boundaries as well as in different scenes.

6 Acknowledgements

This work is in part supported by Guangxi Science Foundation, P. R. of China (GKJ03429-3).

References

[Bra98] Gary R. Bradski. Real time face and object tracking as a component of a perceptual user interface. In *Proceedings of the 4th IEEE Workshop on Applications of Computer Vision (WACV'98)*, page 214. IEEE Computer Society, 1998.

[XP97] Chenyang Xu and Jerry L. Prince. Gradient vector fbw: A new external force for snakes. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pages 66–72. IEEE Computer Society, 1997.