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

CSRN 2902

(Eds.)

- **Vaclav Skala**
  University of West Bohemia, Czech Republic

Computer Science Research Notes

27. International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision
WSCG 2019
Plzen, Czech Republic
May 27 – 30, 2019

Proceedings

WSCG 2019

Proceedings – Part II

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

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

CSRN 2902

(Eds.)

- **Vaclav Skala**
  University of West Bohemia, Czech Republic

Computer Science Research Notes

27. International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision
WSCG 2019
Plzen, Czech Republic
May 27 – 30, 2019

Proceedings

WSCG 2019

Proceedings – Part II

Vaclav Skala - UNION Agency

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

Computer Science Research Notes
CSRN 2902

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

Managing Editor: Vaclav Skala

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

Published in cooperation with the University of West Bohemia
Univerzitní 8, 306 14 Pilsen, Czech Republic

ISSN 2464-4617 (Print) ***ISSN 2464-4625*** (CD/DVD)
ISBN 978-80-86943-38-1 (CD/-ROM)

WSCG 2019

International Program Committee

Anderson,M. (Brazil)
Baranoski,G. (Canada)
Benes,B. (USA)
Benger,W. (Austria)
Bilbao,J. (Spain)
Bouatouch,K. (France)
Bourke,P. (Australia)
Coquillart,S. (France)
Dachsbacher,C. (USA)
Durikovic,R. (Slovakia)
Feito,F. (Spain)
Ferguson,S. (United Kingdom)
Galo,M. (Brazil)
Gavrilova,M. (Canada)
Giannini,F. (Italy)
Gudukbay,U. (Turkey)
Guthe,M. (Germany)
Chaudhuri,D. (India)
Chover,M. (Spain)
Jeschke,S. (Austria)
Juan,M. (Spain)
Klosowski,J. (USA)
Lee,J. (USA)
Liu,S. (China)
Lobachev,O. (Germany)
Manoharan,P. (India)
Max,N. (USA)
Molla,R. (Spain)
Montrucchio,B. (Italy)
Muller,H. (Germany)
Pan,R. (China)
Paquette,E. (Canada)
Pedrini,H. (Brazil)
Richardson,J. (USA)
Rojas-Sola,J. (Spain)
Sarfraz,M. (Kuwait)
Savchenko,V. (Japan)
Segura,R. (Spain)
Skala,V. (Czech Republic)
Sousa,A. (Portugal)
Stuerzlinger,W. (Canada)
Szecsi,L. (Hungary)
Thalmann,D. (Switzerland)
Tokuta,A. (USA)
Torrens,F. (Spain)
Trapp,M. (Germany)
Wuensche,B. (New Zealand)
Wuethrich,C. (Germany)
Yin,Y. (USA)

WSCG 2019

Board of Reviewers

Anderson,M. (Brazil)
Assarsson,U. (Sweden)
Ayala,D. (Spain)
Baranoski,G. (Canada)
Benes,B. (USA)
Benger,W. (Austria)
Bilbao,J. (Spain)
Birra,F. (Portugal)
Bouatouch,K. (France)
Boukaz,S. (France)
Bourke,P. (Australia)
Cakmak,H. (Germany)
Carmo,M. (Portugal)
Carvalho,M. (Brazil)
Coquillart,S. (France)
Dachsbacher,C. (USA)
De Martino,J. (Brazil)
Durikovic,R. (Slovakia)
Eisemann,M. (Germany)
Feito,F. (Spain)
Ferguson,S. (United Kingdom)
Galo,M. (Brazil)
Gavrilova,M. (Canada)
Gdawiec,K. (Poland)
Gerhards,J. (France)
Giannini,F. (Italy)
Goncalves,A. (Portugal)
Gudukbay,U. (Turkey)
Guthe,M. (Germany)
Hansford,D. (USA)
Hermosilla Casajus,P. (Germany)
Chaudhuri,D. (India)
Chover,M. (Spain)
Ivrissimtzis,I. (United Kingdom)
Jeschke,S. (Austria)
Joan-Arinyo,R. (Andorra)
Jones,M. (United Kingdom)
Juan,M. (Spain)
Kanai,T. (Japan)
Klosowski,J. (USA)
Komorowski,J. (Poland)
Krueger,J. (Germany)
Kurt,M. (Turkey)
Kyratzi,S. (Greece)
Larboulette,C. (France)
Lee,J. (USA)
Lisowska,A. (Poland)
Liu,S. (China)
Lobachev,O. (Germany)
Manoharan,P. (India)
Manzke,M. (Ireland)
Mastermayer,A. (Germany)
Max,N. (USA)
Meyer,A. (France)
Mohd Suaib,N. (Malaysia)
Molla,R. (Spain)
Montrucchio,B. (Italy)

Muller,H. (Germany)
Nishio,K. (Japan)
Oliveira,J. (Portugal)
Oyarzun Laura,C. (Germany)
Pan,R. (China)
Papaioannou,G. (Greece)
Paquette,E. (Canada)
Pasko,A. (United Kingdom)
Pedrini,H. (Brazil)
Ramires Fernandes,A. (Portugal)
Richardson,J. (USA)
Rodrigues,N. (Portugal)
Rojas-Sola,J. (Spain)
Sarfraz,M. (Kuwait)
Savchenko,V. (Japan)
Segura,R. (Spain)
Shum,H. (United Kingdom)
Sik-Lanyi,C. (Hungary)
Skala,V. (Czech Republic)
Sousa,A. (Portugal)
Stuerzlinger,W. (Canada)
Szecsi,L. (Hungary)
Thalmann,D. (Switzerland)
Todt,E. (Brazil)
Tokuta,A. (USA)
Toler-Franklin,C. (USA)
Torrens,F. (Spain)
Trapp,M. (Germany)
Tytkowski,K. (Poland)
Vanderhaeghe,D. (France)
Weber,A. (Germany)
Wuensche,B. (New Zealand)
Wuethrich,C. (Germany)
Cai,Y. (Singapore)
Yin,Y. (USA)
Yoshizawa,S. (Japan)
Zwettler,G. (Austria)

CSRN 2902

WSCG 2019 Proceedings
Part II

Contents

ISSN 2464-4617 (print) Computer Science Research Notes
ISSN 2464-4625 (DVD) CSRN 2901 WSCG Proceedings Part II

Tools for development of interactive web-based maps: application in healthcare

Matěj Karolyi
Faculty of Informatics, Masaryk University
Botanická 68a
602 00, Brno, Czech Republic
karolyi@iba.muni.cz

Jan Krejčí
Department of Telecommunications, Brno University of Technology
Technická 10
616 00, Brno, Czech Republic
krejci@iba.muni.cz

Jakub Ščavnický
Institute of Health Information and Statistics of the Czech Republic and Institute of Biostatistics and Analyses, Faculty of Medicine, Masaryk University - joint workplace
Palackého nám. 4,
128 01, Praha 2, Czech Republic
scavnicky@iba.muni.cz

Roman Vyškovský
Institute of Biostatistics and Analyses, Faculty of Medicine, Masaryk University
Kamenice 3,
625 00, Brno, Czech Republic
vyskovsky@iba.muni.cz

Martin Komenda
Institute of Health Information and Statistics of the Czech Republic and Institute of Biostatistics and Analyses, Faculty of Medicine, Masaryk University - joint workplace
Palackého nám. 4,
128 01, Praha 2, Czech Republic
komenda@iba.muni.cz

ABSTRACT

Interactive visualisations on the Internet have become commonplace in recent years. Based on such publicly available visualisations, users can obtain information from various domains quickly and easily. A location-specific method of data presentation can be much more effective using map visualisation than using traditional methods of data visualisation, such as tables or graphs. This paper presents one of the possible ways of creating map visualisations in a modern web environment. In particular, we introduce the technologies used in our case together with their detailed configuration. This description can then serve as a guide for the customisation of the server environment and application settings so that it is easy to create the described type of visualisation outputs. Together with this manual, specific cases are presented on the example of an application which was developed to display the location of medical equipment in the Czech Republic based on data collected from healthcare providers.

Keywords

medical equipment, data analysis, map visualisation, healthcare, Czech Republic.

1. INTRODUCTION

Data visualisation is a huge interdisciplinary domain that has reached mainstream consciousness: an ever increasing number of not only professionals, but also academic, government and business organisations have become interested and involved in this matter [Kirk16]. Proper data visualisation requires the understanding of input data, which can often be rather complex, combined with an analytical point of view of data processing as well as the interpretation by end users. Each new opportunity for data visualisation represents a new and unique presentation of a certain data set. Development and implementation of any innovative and interactive web-based project of data visualisation is quite a challenging issue in terms of smooth and effective delivery of presented information to selected target groups of users [Murr17]. The main goal should always be to help users explore available data sets based on different views and allow them to choose their own ways of obtaining customised data subsets. Geographic maps are one of the most commonly used approaches for data visualisation. It has been used as a metaphor for visualisation, which produce a more readable form for those people who are familiar with reading maps [BYPA15].

With the proliferation of information graphics and online tools that create interactive data visualisations, combined with research that shows the effectiveness of including visuals in medical and healthcare fields, there is an increased urgency to determine effective practices for their creation and use [MeWa17]. In this particular domain, many modern and interesting projects aspire to present their data sets using web-

based map visualisations to make high-quality information available to anyone interested. With the growing trend of health data production among all involved stakeholders, data visualisation applications have become very popular in the last decade. Several recent examples are briefly introduced below. The Institute for Health Metrics and Evaluation[1] aims to improve the health of the world's populations by providing the best information on population health. UNICEF and the World Health Organisation[2] have together developed a web portal containing all available data and the latest child mortality estimates for each country. The Health Intelligence project[3] highlights applications on how data can be used, transformed to meaningful information and subsequently communicated online. It represents an integrated set of data, methods and processes, tools and individuals working together in order to turn health data in insights and actionable information, converting information into evidence and knowledge, and ultimately communicating findings, results, and key messages to all those who need them.

Purpose of the Map of Medical Equipment

The joint workplace of the Institute of Health Information and Statistics (IHIS) of the Czech Republic and the Institute of Biostatistics and Analyses (IBA), Faculty of Medicine, Masaryk University has been producing a large amount of valuable analyses, reports and publications intended for the general public, representatives of regional authorities as well as healthcare providers. Selected results are also presented online in the form of interactive data browsers (dynamic presentation of data) and slideshows presentation tools (static presentation of data).

The Map of Medical Equipment (MME), which is available at https://ztnemocnice.uzis.cz, is a pilot representative of an interactive web-based application aimed at displaying the availability of different categories of medical equipment in individual healthcare providers in the Czech Republic. The map was developed under the guarantee of the joint workplace of IHIS and IBA. The use of all the developed map visualisation has the significant advantage that most people understand geographic maps easily thanks to their early exposure to maps in schools. In nowadays in no such as application in Czech governmental sphere which shows locations of bought medical equipment by healthcare providers. Therefore, this tool is useful for checking this kind of information by governmental workers, healthcare facilities' stakeholders and citizens.

2. METHODS

The IHIS is responsible for the development of the platform as well as for the preparation of underlying data. As part of the development, the application is continuously automatically deployed on development servers, where their internal review can be carried out by a wider group of workers and guarantors. After approving individual bundles of changes, it is then deployed to IHIS production servers. The deployment process is partially automated using the Jenkins automation server [Jenk00].

Deployment environment

The deployment environment consists of four mutually dependent instances: (i) a development webserver with a continuous integration system, (ii) a production webserver, (iii) an application server with a distributed version control system and (iv) an application server with map sources. All of the above-mentioned instances are scalable virtual machines located on the same physical machine. An Ubuntu Server 16.04 with Long Term Support (LTS) with the application packages described in Section 3.1 provides the environment for running these servers.

Data layer

The data upon which the application is built are stored in a PostgreSQL 9.5 database system [Momj01, Post00]. The database was created in the Entity first way that can be performed with services provided by the Doctrine2 ORM framework [Dung13]. The database and public schema were created using the Symfony commands doctrine:database:create and doctrine:schema:create while the meta information about the database was taken from the symfony parameters.yml config file. After the entities were created, another command (doctrine:schema:update) was used to create database tables and sequences necessary to prepare the database for data import. This time, meta information was taken from the attribute annotation in the entity.

All steps necessary to make the data cleaned and prepared were performed in R language[4], version 3.4.3. The tables were saved as csv files in UTF-8 encoding and the import itself was carried out using the pgAdmin III import tool. This tool also makes it possible to reimport the whole data set or import new data rows easily without the need for writing insert commands in cases when new data are delivered to the web development department.

[1] http://www.healthdata.org

[2] http://www.childmortality.org

[3] http://www.publichealthintelligence.org

[4] https://www.r-project.org/

ISBN 978-80-86943-38-1

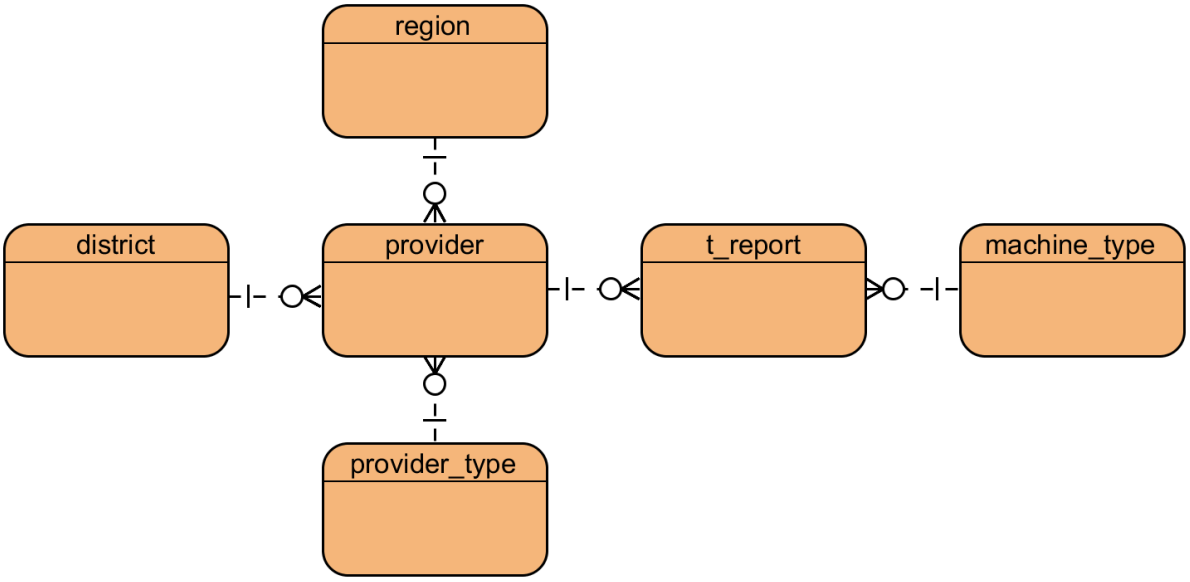


Figure 1: Conceptual model of database entities

The MME application is based on a simple database schema that relies mainly on two tables (conceptual model in Figure 1). The description of all healthcare facilities (hospitals, physicians, dentists etc.) providing any medical equipment is stored in the "provider" table. The reports containing the number and type of medical equipment provided by medical facilities are stored in the "t_report" table.

The schema is complemented by listings carrying the information of machine type, provider type, region and district.

Data sources

Data were obtained from two different sources: the National Register of Healthcare Providers[5] and the Annual Report on Medical Equipment of a Healthcare Facility [Výka00]. The former contains a list of medical facilities; besides general information (such as the address and type of facility), GPS coordinates are also available, and these can be used to display the provider on a map. The latter source includes reports on equipment that must compulsorily be completed each year by all healthcare providers.

Data preparation for visualisation with Leaflet

Leaflet is an open-source library written in Javascript. It is used for the development of interactive maps within responsive web applications and portals. It works efficiently across all major desktop and mobile platforms. The core of Leaflet can be extended with various plugins (such as routing features, heatmaps, geocoding etc.) and weighs just about 38 kB of Javascript code which is free to extend thanks to the open licensing [Leaf00].

Initialisation of a Leaflet map requires tile layer specification. There are many options for Leaflet map providers (see the official site at https://leafletjs.com/). We have used the OpenStreetMap tile layer in MME. After tiles are defined, initial map zoom, view and graphical points can be added. Each point requires GPS coordinates to be drawn. Furthermore, points may carry custom information in terms of their icon, colour or tooltip. The Leaflet map then becomes an interactive map that allows users to zoom in and out, move around a specific tile layer, click on specific points and display the included information.

Types of views

MME provides several views of the data. Users can get a comprehensive picture of the latest reported situation on medical equipment either across the Czech Republic or in individual regions or cities. Data selection can be refined by choosing a specific type of equipment or a specific region. The primary visualisation element is a map layer with an extra layer of points: each point symbolises a particular provider. If more than one healthcare providers are located at the same address (and therefore in the same coordinates), the point is shared. Details of individual healthcare providers are also available, combined with all necessary information (identification, name of institution, address, list of medical equipment).

[5] http://nrpzs.uzis.cz/

ISBN 978-80-86943-38-1

Last but not least, the search of the nearest providers with a certain type of technique is available based on the address entered and with optional navigation support. Some of views are described in the Results chapter.

Geocoding with Nominatim

Nominatim is a search engine for OpenStreetMap (OSM) data sources and can be generally described as a reverse geocoder for OSM data: it searches OSM data by name and address and generates synthetic addresses of OSM points. The tool is maintained under the General Public Licence (GPL) v2 licence and can be used as an own hosted installation (our use-case described as follows) or free service hosted[6].

2.1.1 Nominatim installation

Nominatim is supported by Linux only, as a package for Ubuntu or Centos distributions or as a docker container [What00]. In case of own installation, it is important to follow the package requirements. Nominatim is dependent on the following general packages: (i) Python (version 3.0 or higher), (ii) PostgreSQL (version 9.1 or higher), (iii) PHP (version 5.4 or higher) and (iv) Apache or Nginx webserver. In addition, Nominatim requires some extra packages for its full operation: (i) GIS spatial DB backend for PostgreSQL - PostGIS (version 2.0 or higher) for storing structured OSM data and (ii) Osmium library for Python – PyOsmium, which maintain OSM data source up-to-date.

Hardware requirements for Nominatim depends on both the size and type of imported map sources. Most frequently, there are two types and sources of the maps: (i) original maps sources in the XML format are available on the OSM website[7] or (ii) data extracts in OSM.PBF format[8] are used. For our purpose, we chose data extracts for a number of reasons: (i) the package contains raw OSM data, (ii) the package is fully suitable for Nominatim, (iii) our data are 100% pure and unfiltered, (iv) the package contains all OSM metadata, (v) the packages are updated several times a day, (vi) our data are GDPR compliant, (vii) additional packages for Wikipedia rankings and Points of Interests are available and (viii) the package can be downloaded free of charge.

Maps can be imported for the entire Earth, for a selected continent or for a set of countries. Package sizes for example in our use-case are typically as follows: (i) 60GB for the entire Earth, (ii) 19GB for Europe or (iii) about 700MB for data source. General requirements (the interval mentioned below is defined from an import for one small country, such as the Czech Republic, an import for the entire Earth) are: (i) CPU from 2 to 6 cores with HyperThreading, (ii) RAM from 4GB to 32GB and (iii) HDD from 30GB to 1TB disk space. Data sources in the OSM format are compressed and after their import into the database, it will take more space and consume lot of computational time for the process of import. For example, data sources for the Czech Republic take up 700MB as source in a OSM file and 12GB as raw data PostgreSQL database; their import takes 6 hours on SSD and 11 on mechanical HDD for an allocated four core virtual server with 8GB RAM.

Address resolving with Nominatim's API

Address resolving can be carried out in two ways: (i) via a graphical user interface (GUI), provided by a Nominatim page running on the web server, see Figure2 and Figure 3, or (ii) via an application

programming interface (API) using the standard HTTP methods.

The API is a tool used to search through OpenStreetMap data either by name and address (the user begins with the address and an appropriate geographic coordinate is returned) or by reverse geocoding (the user begins with a geographic coordinate and the nearest known address is returned).

The example query for resolving the address "Postovska3, Brno" geographic coordinate with request for address details; the XML output can be as follows:

https://nominatim.openstreetmap.org/search?q=3+postovska+brno&format=xml&polygon=1&addressdetails=1

For this query, the response shown in Figure 4 is returned.

Nominatim API supports national abbreviations for common short-term names for street and square terms and supports national and regional languages sets, queried as UTF-8 search strings.

3. Results

In this section, we describe the web-based application providing a complex overview and an advanced searching feature using interactive web-based maps. The first part is focused on the background of the whole application and should summarise all tools needed for the creation of such an application. The second part describes a specific part of MME from the user's point of view.

[6] https://nominatim.openstreetmap.org/

[7] https://planet.openstreetmap.org/

[8] https://download.geofabrik.de/

ISBN 978-80-86943-38-1

Technical description of the toolkit

This paper summarises libraries, frameworks and operational system environment settings which are needed for the creation of interactive web-based map visualisations like these in MME.

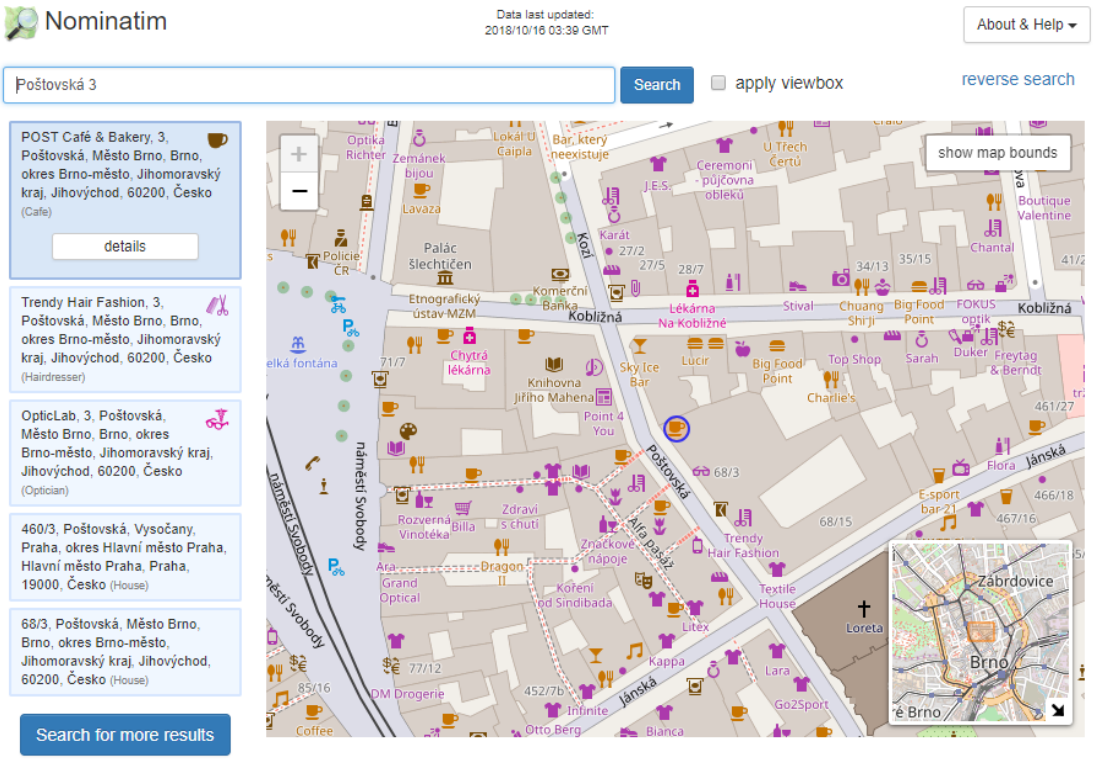


Figure 2: Nominatim query via GUI (search engine)

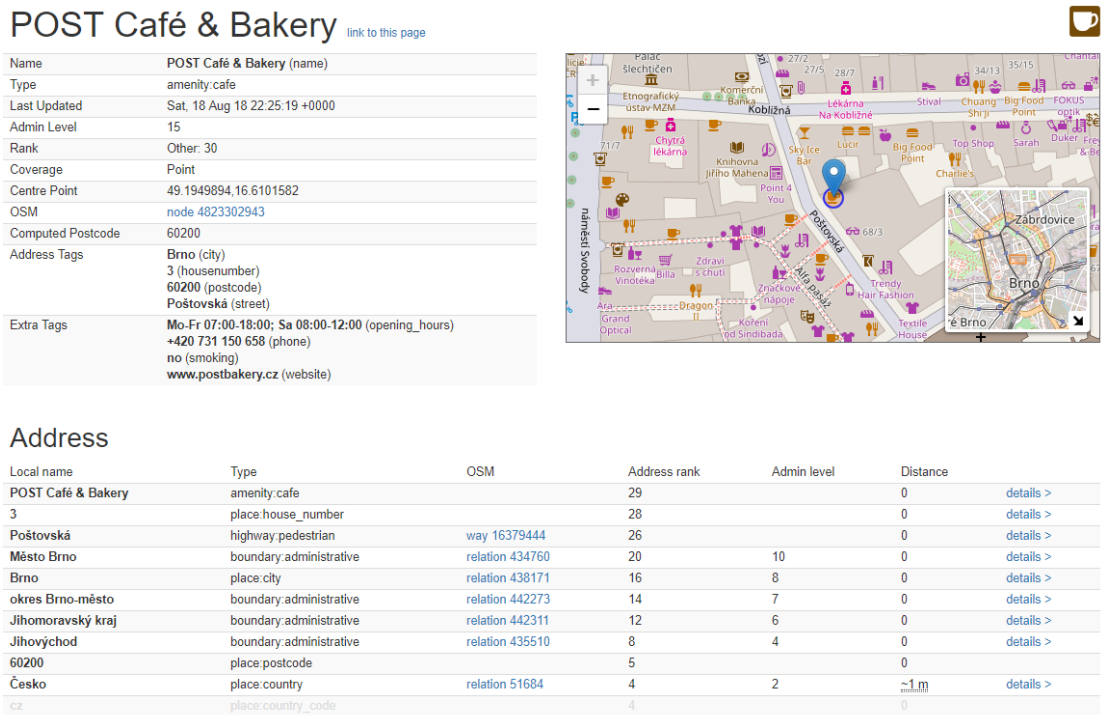


Figure 3: Nominatim query via GUI interface (location detail)

ISBN 978-80-86943-38-1

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<searchresults more_url="https://nominatim.openstreetmap.org/search.php?
q=3+postovska+brno&addressdetails=1&polygon=1&exclude_place_ids=60070363%2C53679170%2C54959027%2C965173&format=xml&accept-
language=cs-CZ" exclude_place_ids="60070363,53679170,54959027,965173" polygon="true" querystring="3 postovska brno" attribution="Data ©
OpenStreetMap contributors, ODbL 1.0. http://www.openstreetmap.org/copyright" timestamp="Tue, 16 Oct 18 03:48:41 +0000">
  <place class="amenity" importance="0.211" type="cafe" display_name="POST Café & Bakery, 3, Poštovská, Město Brno, Brno, okres Brno-město,
  Jihomoravský kraj, Jihovýchod, 60200, Česko" lon="16.6101582" lat="49.1949894" polygonpoints="[[16.6101582,49.1950394],
  [16.61019355533906,49.195024755339055],[16.610208200000002,49.1949894],[16.61019355533906,49.19495404466094],
  [16.6101582,49.194939399999996],[16.610122844660943,49.19495404466094],[16.6101082,49.1949894],
  [16.610122844660943,49.195024755339055]]" boundingbox="49.1949394,49.1950394,16.6101082,16.6102082" place_rank="30" osm_id="4823302943"
  osm_type="node" place_id="60070363" icon="https://nominatim.openstreetmap.org/images/mapicons/food_cafe.p.20.png">
    <cafe>POST Café & Bakery</cafe>
    <house_number>3</house_number>
    <pedestrian>Poštovská</pedestrian>
    <suburb>Město Brno</suburb>
    <city>Brno</city>
    <county>okres Brno-město</county>
    <state>Jihovýchod</state>
    <postcode>60200</postcode>
    <country>Česko</country>
    <country_code>cz</country_code>
  </place>
  <place class="shop" importance="0.211" type="hairdresser" display_name="Trendy Hair Fashion, 3, Poštovská, Město Brno, Brno, okres Brno-město,
  Jihomoravský kraj, Jihovýchod, 60200, Česko" lon="16.6104506" lat="49.1947341" polygonpoints="[[16.6104506,49.1947841],
```


Figure 4: Nominatim query via an API interface (returned XML data for an example query)

Of course, that this is not the only approach possible and there could certainly be different equivalents; this is just an example shown on our specific case. The following list shows individual parts of the toolkit, divided into several categories:

- **Operation system environment**: Ubuntu 16.04, PHP 7.0, Python 3.0
- **Persistent layer**: PostgreSQL 9.5, Doctrine DBAL 2.8
- **Application's backend**: Symfony 3.4 framework for PHP. Doctrine ORM 2.5, Twig template engine 2.0
- **Frontend and user interface**: Leaflet 1.3, Select2 4.0, jQuery 2.2.4, d3.js 3.5

Use case: Finding the nearest healthcare provider

Based on this view of healthcare providers and their reported medical equipment, the user is able to find the closest providers with specific medical equipment quickly and easily. The typical user workflow is as follows:

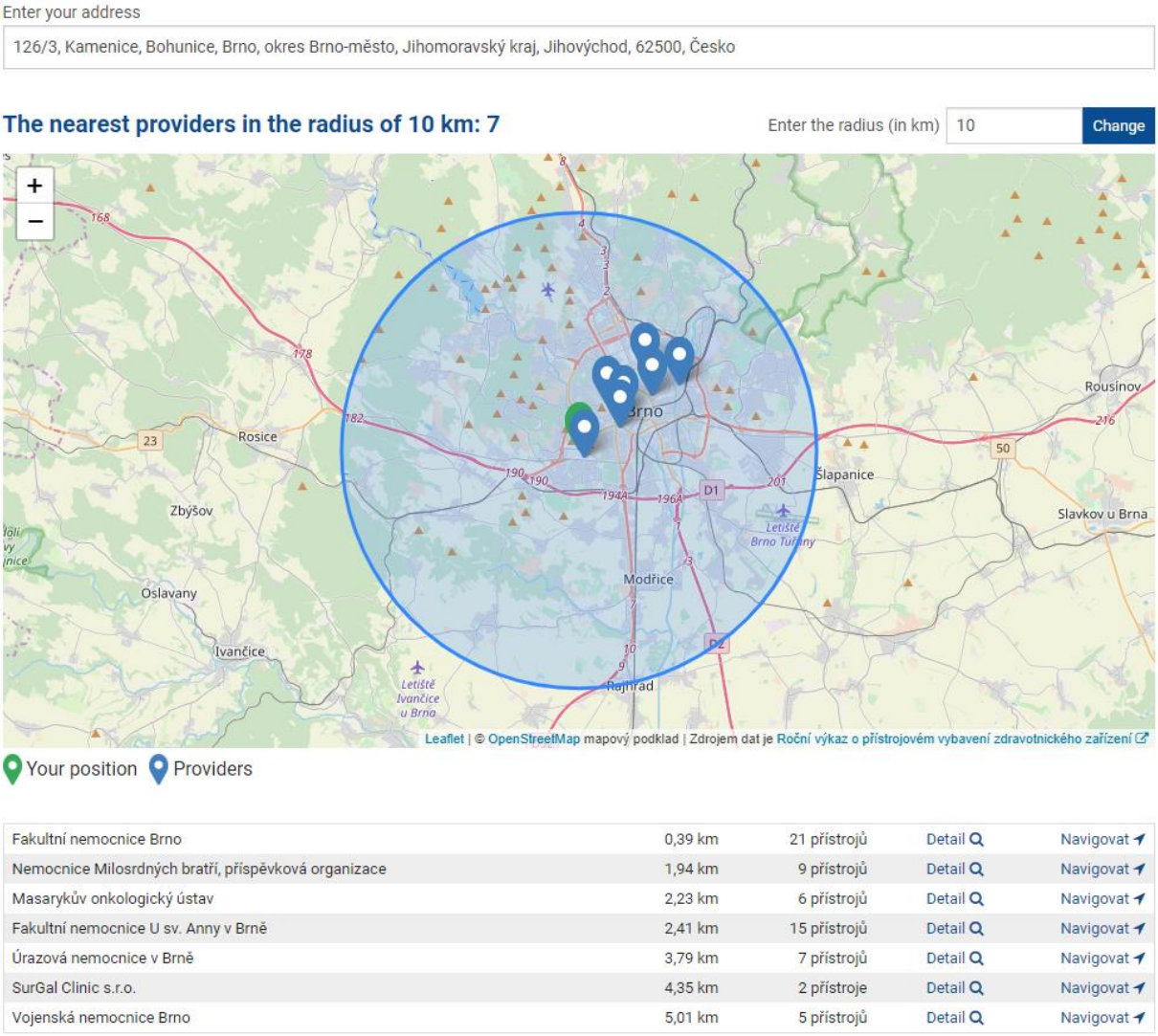
- The user chooses which category of medical equipment or type of healthcare provider is interesting for him/her.
- The user then types the address he/she wants to find.
- The portal returns the list of suggested addresses that match the given expression.
- The user chooses one of the suggested addresses.
- The position of the chosen address is plotted on the map and a circle of ten-kilometre radius is highlighted from this address. All providers within a radius that meet the conditions selected in point 1 are displayed on the map and listed in a table below the map (see Figure 5).
- It is possible to increase or decrease the size of the above-mentioned radius by typing a specific value above the map.
- The user can then view the details of one of the listed providers or start navigating from the selected address to a healthcare provider.

4. Discussion and future work

The MME interactive portal provides a clear and comprehensive information about 15,104 pieces of medical equipment divided into 76 categories from a total of 209 healthcare providers, which are located in 14 regions in terms of global geographical overview and accessibility. The published data refer to all providers of inpatient care who have completed the obligatory Annual Report on Medical Equipment. The currently presented information (January 2019) on medical equipment is valid as of December 31, 2017. The user can choose among four ways of getting a detailed map of medical equipment of his/her interest: (i) searching by healthcare provider, using either provider name, VAT identification number or provider type, (ii) searching by medical equipment category, (iii) searching by region and district, (iv) accessibility searching by location using address, where all closest providers having selected medical equipment can easily be found in a particular distance radius. The results on a given query are always presented by geographic data in the form of a zoomable map as well as in the form of a data table with orderable columns.

We have created a non-trivial portal consisting of several map visualisations and detailed medical equipment listings. Collected data can be displayed from several different points of view, and individual views are connected with intelligible navigation. We have achieved this result by linking the technologies and tools described in this paper, which were combined together with an appropriate set-up of individual components.

ISBN 978-80-86943-38-1



Fakultní nemocnice Brno	0,39 km	21 přístrojů	Detail	Navigovat
Nemocnice Milosrdných bratří, příspěvková organizace	1,94 km	9 přístrojů	Detail	Navigovat
Masarykův onkologický ústav	2,23 km	6 přístrojů	Detail	Navigovat
Fakultní nemocnice U sv. Anny v Brně	2,41 km	15 přístrojů	Detail	Navigovat
Úrazová nemocnice v Brně	3,79 km	7 přístrojů	Detail	Navigovat
SurGal Clinic s.r.o.	4,35 km	2 přístroje	Detail	Navigovat
Vojenská nemocnice Brno	5,01 km	5 přístrojů	Detail	Navigovat

Figure 5: Search of the nearest healthcare providers

During the entire period of application run up to now, we have not experienced any long-term service outages or performance fluctuations, not even during the moments of automatic daily updates of a persistent layer from an external resource containing information about healthcare providers (National Registry of Healthcare Providers). In order to present a complete dataset on our MME, import of all healthcare providers records from across the Czech Republic will definitely be the next step forward. This improvement will follow essential changes in the Czech legislation, where the Ministry of Health of the Czech Republic is the leading competent authority.

5. REFERENCES

[BYPA15] BIUK-AGHAI, ROBERT P. ; YANG, MUYE ; PANG, PATRICK CHEONG-IAO ; AO, WAI HOU ; FONG, SIMON ; SI, YAIN-WHAR: A map-like visualisation method based on liquid modelling. In: *Journal of Visual Languages & Computing* Bd. 31 (2015), S. 87–103

[Dung13] DUNGLAS, KÉVIN: *Persistence in PHP with the Doctrine ORM* : Packt Publishing Ltd, 2013. — Google-Books-ID: 0NtdAgAAQBAJ — ISBN 978-1-78216-411-1

[Jenk00] *Jenkins*. URL https://jenkins.io/index.html. - abgerufen am 2018-11-23. — Jenkins

[Kirk16] KIRK, ANDY: *Data Visualisation: A Handbook for Data Driven Design* : SAGE, 2016. — Google-Books-ID: wNpsDAAAQBAJ — ISBN 978-1-4739-6631-4

[Leaf00] *Leaflet — an open-source JavaScript library for interactive maps*. URL https://leafletjs.com/. - abgerufen am 2018-11-23

ISBN 978-80-86943-38-1

[MeWa17] MELONCON, L. ; WARNER, E.: Data visualizations: A literature review and opportunities for technical and professional communication. In: *2017 IEEE International Professional Communication Conference (ProComm)*, 2017, S. 1–9

[Momj01] MOMJIAN, BRUCE: *PostgreSQL: introduction and concepts*. Bd. 192 : Addison-Wesley New York, 2001

[Murr17] MURRAY, SCOTT: *Interactive Data Visualization for the Web: An Introduction to Designing with* : O'Reilly Media, Inc., 2017. — Google-Books-ID: NGwvDwAAQBAJ — ISBN 978-1-4919-2132-6

[Post00] *PostgreSQL: About*. URL https://www.postgresql.org/about/. - abgerufen am 2018-11-23

[Výka00] *Výkazy na rok 2017 | ÚZIS ČR*. URL http://www.uzis.cz/vykazy/vykazy-rok-2017#T. - abgerufen am 2018-11-23

[What00] *What is a Container*. URL https://www.docker.com/resources/what-container. - abgerufen am 2018-11-23

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

Porting A Visual Inertial SLAM Algorithm To Android Devices

Jannis Möller
Technische Hochschule Köln
Steinmüllerallee 1
51643, Gummersbach,
Germany
jmoeller@th-koeln.de

Benjamin Meyer
Technische Hochschule Köln
Steinmüllerallee 1
51643, Gummersbach,
Germany
benjamin.meyer@th-koeln.de

Martin Eisemann
Technische Hochschule Köln
Steinmüllerallee 1
51643, Gummersbach,
Germany
martin.eisemann@th-koeln.de

ABSTRACT

Simultaneous Localization and Mapping aims to identify the current position of an agent and to map his surroundings at the same time. Visual inertial SLAM algorithms use input from visual and motion sensors for this task. Since modern smartphones are equipped with both needed sensors, using VI-SLAM applications becomes feasible, with Augmented Reality being one of the most promising application areas. Android, having the largest market share of all mobile operating systems, is of special interest as the target platform. For iOS there already exists a high-quality open source implementation for VI-SLAM: The framework VINS-Mobile. In this work we discuss what steps are necessary for porting it to the Android operating system. We provide a practical guide to the main challenge: The correct calibration of device specific parameters for any Android smartphone. We present our results using the Samsung Galaxy S7 and show further improvement possibilities.

Keywords

Augmented Reality; Visual Inertial SLAM; Smartphones; Camera calibration

1 INTRODUCTION

Visual inertial SLAM describes a class of algorithms that use both visual and inertial measurements to build a map of the environment and at the same time locate an agent in it. Besides the research field of robotics, they also are of interest for Augmented Reality (AR) applications. In this application context, the goal is the perspectively correct rendering of virtual 3D content for a given viewing position, either using see-through devices like the Microsoft HoloLens or on top of a live recorded camera image.

Recently, computing hardware has been getting more and more powerful. The smartphone market in particular is developing very fast. This rapid progress makes algorithms that required powerful desktop hardware a few years ago now suitable for use on mobile devices.

In order to enable a robust and therefore immersive augmented reality experience on the users' devices, three-dimensional localization within the environment is required. However, most freely available augmented reality frameworks only offer tracking of specific markers. Since all popular smartphones are equipped with a camera and an inertial measurement unit (IMU) these days, visual inertial odometry and SLAM poses a suitable solution for this problem. To utilize these algorithms a set of device specific parameters has to be known beforehand.

The intention behind this work is to give a simple explanation of the steps that are required to port an existing framework to Android. We focus especially on providing a comprehensible and practical guide for determining these parameters. Less relevant parts, like translating the code base, are omitted for the sake of clarity as they are highly depended on the individual framework. The procedures are exemplarily explained on the basis of the framework *VINS Mobile* [LQH*17], which is originally only available for iOS and has been ported to Android in the course of this work. However, the procedure of porting and calibrating the parameters are of relevance for other algorithms or frameworks as well.

This work is divided into six sections:
Section 2 gives an overview over the recent development in the field of Augmented Reality frameworks as well as the research field of visual inertial SLAM algorithms. It also provides a brief introduction into the algorithm behind the VINS-Mobile framework. In Section 3 we show the different steps that are required to calibrate the device specific parameters. Section 4 ex-

plains the main problems we encountered in regards to Android hard- and software. We present and analyze the results of our quantitative and qualitative tests in Section 5. Finally, in Section 6 we list options of further improvements for future work.

2 RELATED WORK

There are many different ready-to-use solutions in the field of augmented reality applications and frameworks. The favored open source candidate for arbitrary AR-projects seems to be the AR framework *ARToolKit* [ARTa]. Its compatibility with Unity and the capability to develop a single application for multiple platforms simultaneously surpass the capabilities of most other open source frameworks. In the current version, however, it offers no possibility to recognize or track a 3D environment using feature matching. It is limited exclusively to 2D images or markers. Future versions of this framework are said to improve feature tracking and add area recognition. But the official development seems to have come to a standstill, after it was bought by *DAQRI*. In the future, development is planned to be continued as the new project *artoolkitX* [ARTb] by the open source community. Compared to this framework other open source solutions are even more limited in scope and function; some offer nothing more than a simple abstraction layer around the image processing library *OpenCV* [OPE].

Furthermore, some proprietary solutions are available, but they are not completely open-sourced and often come at a high financial price in later use. Some of these are *Vuforia* [VUF], *Kudan* [KUD], *MAXST* [MAX] and *Wikitude* [WIK], as well as the free to use *ARKit* [ARK] from Apple and *ARCore* [ARC] from Google. It has to be mentioned that recent advancements in these proprietary frameworks included the implementation of more complex localization and tracking algorithms. These features are often called *markerless tracking*, or just *motion tracking*. In addition, some of the frameworks like ARKit and ARCore also support basic understanding of the scene through plane detection. Even though being rich in features, the fact that they are either very costly or not completely open-source is a knockout criterion for when adaptability is a high priority like in a research context.

In the general field of visual inertial odometry and SLAM algorithms, there are many approaches often coming with open source implementations. They can be categorized into filter-based approaches, such as [LM13, HKBR14, BOHS15] and optimization-based approaches such as [IWKD13, SMK15, LLB*15]. Filter-based ones usually need less computational resources and optimization-based ones might provide greater accuracy. However, most of these are designed or implemented for other platforms with different hardware requirements, the calculations are often done offline, not in real time or on powerful desktop hardware.

One of the few mobile solutions is *VINS-Mobile* [LQH*17], the iOS port of the *VINS-Mono* project [QLS17]. The source code and the underlying scientific papers are publicly available [VIN]. It fuses visual and inertial measurements in a tightly-coupled, optimization-based approach. The position and orientation for successive selected camera frames (keyframes) are stored in a global pose graph. Several performance improving measures are utilized to enable usage of mobile hardware. The main one being the limitation of the global pose graph optimization to a sliding window of 10 keyframes. Another important feature is the initialization step, which automatically determines the metric scale. There is no need for assumptions or knowledge about the environment, but the device-specific parameters have to be known. To correct the occurring position drift a simple bag of words based loop closure procedure is integrated into the global pose optimization.

3 CALIBRATION

To port a framework many different steps are required. Besides the translation of operating system specific code to Android, a few parameters have to be adjusted to the new hardware platform. The visual inertial SLAM algorithm assumes that these parameters are known before run-time. They include the intrinsic camera parameters, focal length and center of the image needed for visual reconstruction and the extrinsic parameters that describe the transformation from the IMU frame to the camera frame for correctly combining inertial information with the camera motion.
This section exemplary shows the different steps that are required to determine these parameters. It can be used as a guideline to port and use this framework on any Android smartphone. We tested the Android port on a Samsung Galaxy S7.

3.1 Intrinsic Parameters

The required intrinsic parameters are the focal length and the center of the image (principal point), both in pixels in X and Y direction. The focal length specifies the distance between the lens and the focal point, while the principle point specifies the image center in the pinhole camera model. It usually is close to the center of the output image. Pixels as a unit of length are used in relation to the photo sensor pixel size. A visualization of the parameters based on the pinhole camera model is shown in Figure 1.

To determine the intrinsic parameters, we took a video of a checkerboard pattern at a video resolution of 480×640, manually extracted 10 to 20 frames

 ISBN 978-80-86943-38-1

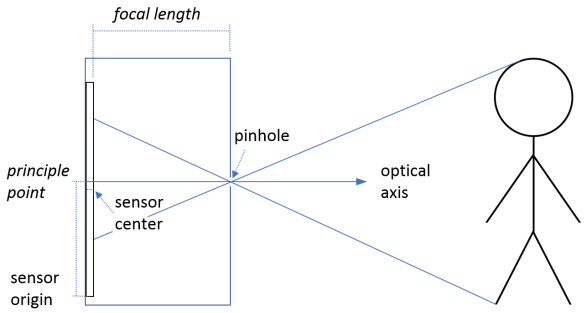


Figure 1: Pinhole camera model with the parameters *focal length* and *principle point*

spanning a variety of different camera positions and reconstructed the intrinsic parameters using *Camera Calibrator* from the software suite *MatLab* [MAT].

The reconstruction of the intrinsic parameters is normally only computable up to a scaling factor. This limitation can be circumvented by manually providing the length of a checkerboard square. Figure 2 shows the mean reprojection error we were able to achieve during the calibration process. Apart from the focal length and principal point this procedure also reconstructs the radial distortion of the lens, which, however, is being ignored in the VINS-Mobile framework.

Instead of using standard checkerboards, it has been proven advantageous to resort to special calibration checkerboards with an odd number of rows and an even number of columns. Unlike with standard checkerboard patterns, orientation is always unique in this combination of row and column count. This will prevent most calibration algorithms from selecting the wrong corner when placing the pattern origin.

During our work, we tested two different setups for this calibration step. The first video was taken with a checkerboard pattern printed on a DinA4 sized sheet of paper. The individual squares had a side length of exactly 30 mm. The mean reprojection error in this setup can be seen in Figure 2(a). For the second video, the filmed pattern was displayed on a flat computer monitor. The size of the squares in this setup was slightly larger at 33.8 mm. The resulting mean reprojection error is shown in Figure 2(b). These two different inputs led to the calibration results printed in Table 1 & 2. As can be seen, the reprojection error from the printed checkerboard is quite large compared to the one from the screen checkerboard. The reason for that is presumably that the printed paper had built up slightly uneven waves due to the large amount of printer ink. The surface not being completely flat caused the error to be more than twice as big. Modern flat-panel displays serve the purpose of a perfect plane reasonably well.

FocalLength:	[498.3953 495.7647]
PrincipalPoint:	[226.4976 319.1671]
RadialDistortion:	[0.2526 -0.5535]
ImageSize:	[640 480]
MeanReprojectionError:	0.4699

Table 1: Camera intrinsics from printed checkerboard

FocalLength:	[478.7620 478.6281]
PrincipalPoint:	[231.9420 319.5268]
RadialDistortion:	[0.2962 -0.6360]
ImageSize:	[640 480]
MeanReprojectionError:	0.2147

Table 2: Camera intrinsics from screen checkerboard

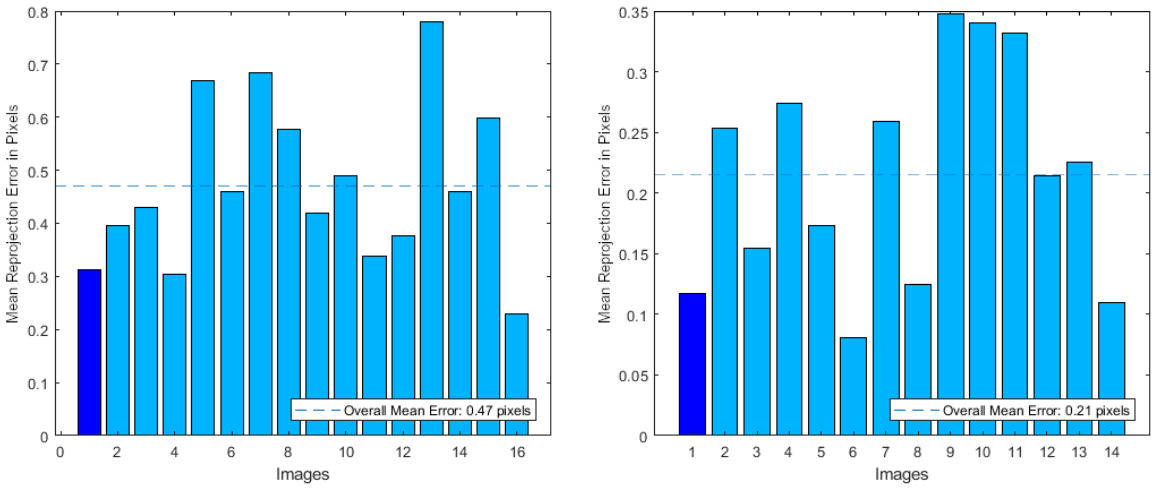
3.2 Extrinsic Parameters

In addition to the described intrinsic parameters, the visual inertial SLAM-algorithm needs specific knowledge about the relative positioning of the inertial measurement unit (IMU) to the camera. The relative orientation is being ignored as the camera and the IMU are expected to be axis-aligned. The extrinsic parameters are highly dependent on the target hardware. In this work, we focus on the Samsung Galaxy S7 as target mobile phone - however, the necessary steps for porting the iOS VINS framework remain the same for arbitrary Android phones.

Usually, the extrinsic parameters are irrelevant in the development of normal smartphone applications, so it is not surprising that the manufacturers did not publish the necessary parameters required for the calibration. Even internet sources that deal with the hardware of smartphones more meticulously have not specified this information at all. As official information about the IMU and camera locations are unavailable, we gathered the needed information from available frontal straight X-ray images [iFi16a] for the Apple iPhone 7 Plus and an open device image with visible PCB [iFi16b] for the Samsung Galaxy S7. Outer dimensions of the smart phones are provided by the manufacturers and can be used as a reference of scale.

In order to make the measurement of the dimensions easier and more precise, the images of the devices were rectified and scaled. The camera respectively the IMU-axis should be the frame of reference for this rectification. A visualization of the results can be seen in Figure 3. Displayed are the iPhone 7 Plus on the left and the Galaxy S7 on the right. Both are frontal views looking straight at the screen. The measurements from the method described above are drawn in green. Blue indicates the values found in the frameworks source code for the iPhone 7 Plus. The translation in direction of the visual axis of the camera is neglected because it cannot be determined easily, is usually very small and thus has only very little impact on the resulting accuracy.

ISBN 978-80-86943-38-1



(a) Printed checkerboard (b) Screen checkerboard

Figure 2: Mean reprojection error from the camera calibration process. The individual bars represent the mean reprojection error of the corresponding images and the dotted line the overall mean reprojection error.

4 PROBLEMS

In the following we describe further hard- and software-posed challenges we encountered.

4.1 Synchronization

On the hardware side of Android, the synchronization of the sensor events needs special attention. It is not guaranteed that acceleration and gyroscope events are always processed alternating, even if set to the exact same frequency. For example, it can happen that ten acceleration events occur at first and the ten corresponding gyroscope events afterwards. Therefore in the porting process, it was necessary to rewrite the buffer system of the IMU measurements. It now uses a separate queue for both sensor event types. The IMU measurements are processed and inserted into the algorithm only once both corresponding measurements occurred. It is also ensured that if one event type skips a time step, the buffer system does not lock forever.

4.2 Unsolved Problems

It was observed that the time difference between the timestamps of the camera and the IMU can be shifted by up to 300 ms. The reason is not obvious and might lie in the hardware and OS-software architecture. At this point it should be noted that the camera image time interval at 30 Hz is just $33.\overline{3}$ ms and the IMU measurement interval at 100 Hz is 10 ms. This shows how severe this desynchronization issue is on Android. A possible explanation for this observation might be that the image buffer of the camera cannot be processed fast enough due to the increasing processing time as the run progresses.

Due to the processing order of the camera and IMU data, the consequences are limited to a general delay in processing. For the preprocessing of the IMU data, it is important that the processing of camera images is deferred longer than the measurement processing of the IMU. If this was not the case, some measurements of the IMU would be lost because they have not yet been inserted into the event queue before the latest image has been processed. For the next picture they would be neglected because their timestamp is older than the one of the last picture. To reliably fix this desynchronization issue the system could be improved by building another buffer system, that fuses both IMU and camera data, before processing them.

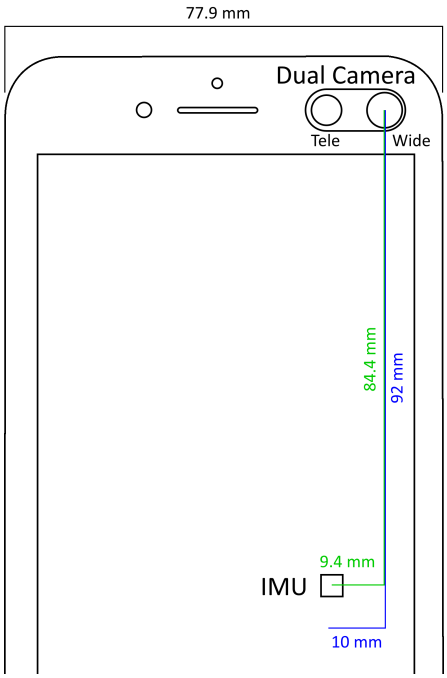
5 RESULTS

After the porting of the framework was completed we compared the Android port with the original iOS version in performance, robustness and accuracy. The results of the quantitative performance study are presented in section 5.1. In section 5.2 we discuss the qualitative differences in the form of an experimental study.

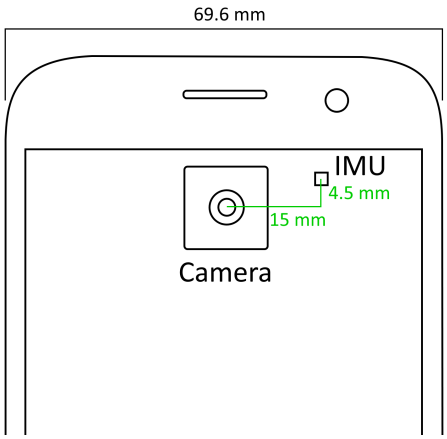
For Android the Samsung Galaxy S7 was used as the test device. It features an Exynos 8890 64-Bit Octa-core processor clocked at 4 x 2.3 GHz and 4 x 1.6 GHz. It has 4 GB of RAM and a 12 MP front-facing camera (f/1.7, 26mm) with optical image stabilization [GSMb]. Thus it can be classified in the middle to upper performance class of Android smartphones.

For iOS the Apple iPad Pro 12.9 (2015) was used for comparison. This tablet features an Apple A9X 64-Bit Dual-core processor clocked at 2.26 GHz. It has 4 GB of RAM too and a 8 MP front-facing camera (f/2.4, 31mm) [GSMa].

ISBN 978-80-86943-38-1



(a) Apple iPhone 7 Plus



(b) Samsung Galaxy S7

Figure 3: Extrinsic parameters: Translation from IMU to Camera. Drawn in blue are the values found in the frameworks source and drawn in green the values determined by our method.

5.1 Quantitative Performance

This section presents the statistical results of our performance benchmark. We gathered timings for each of the individual steps of the algorithm. The resulting statistical observations are listed in Table 3. The structure and values for iOS are inferred from [LQH*17]. Android Studio in combination with the native developement kit (ndk) and its GNU Compiler Collection (GCC) variant was utilized for porting and compiling the framework on Android. To reach comparable levels of performance on Android it is necessary to enable the highest possible compiler optimization level. Without these optimizations the modules relying on the Ceres Solver library are more than a magnitude slower.

T	Module	Freq.	iOS	Android
1	Feature Detection	10 Hz	17 ms	16 ms
	Feature Tracking	30 Hz	3 ms	8 ms
	Visualization	30 Hz		6 ms
2	Initialization	once	80 ms	268 ms
	Nonlinear Opt.	10 Hz	60 ms	113 ms
3	Loop Detection	3 Hz	60 ms	34 ms
	Feature Retrieval	3 Hz	10 ms	44 ms
4	Global Pose Opt.	marg.		29 ms

Table 3: Performance comparison. Modules are grouped by threads (T) they run in.

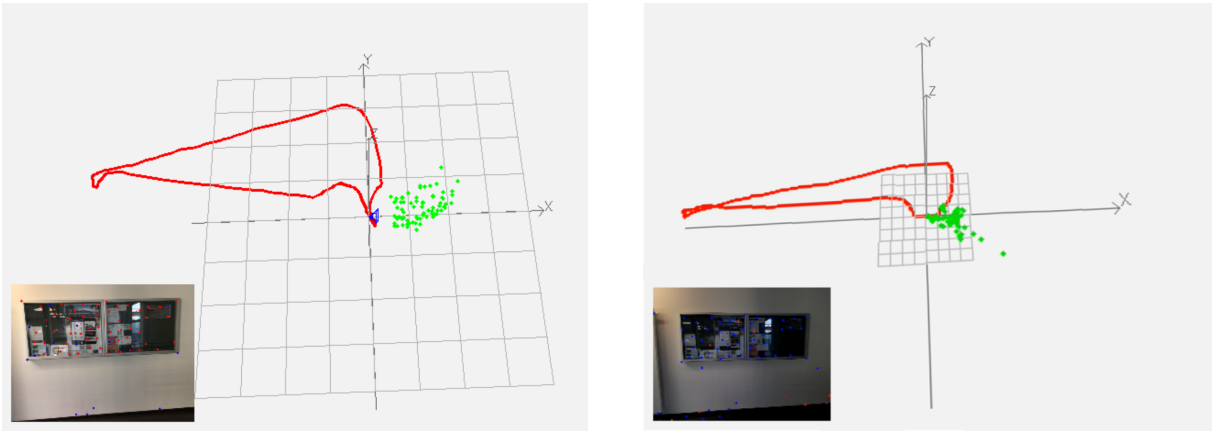
For the measurements of the optimized version we chose two different approaches:
First, we ran 10 tests which we canceled immediately after initialization. These were used solely to gather information about the average initialization time. Second, we carried out 5 more test runs, each limited to 20 seconds. In each of them the initialization was completed and a 360 degree turn performed. In 4 out of 5 of them the previously visited regions were recognized and the loops in the pose graph successfully closed. Like in the first approach, we used the collected data to extract a meaningful average of the other algorithm steps over these runs.

Not specified in table 3 is the total time of the `onImageAvailable` function, which is executed for each camera input frame. It includes the necessary image format conversion as well as the feature detection and tracking and the visualization. As it takes up 37.25 ms on average, the processing isn't able to keep up with the image input rate of 30 Hz and thus causes skipping of some camera frames. The resulting delay and skipping of frames is noticeable to the naked eye, but partly affects the iOS version as well, though not as strongly.

The observed value for loop detection is surprisingly low in comparison to the iOS measurements. This could be attributed to the short duration of the test. As the pose graph continues to grow over time, this module will take more time.

Overall, the results indicate that the interaction between software and hardware on iOS is optimized better. This is particularly noticeable in the measurements of initialization and nonlinear optimization, as both heavily rely on the Ceres Solver library. The Android version would potentially benefit a lot from a better parallelization implementation of the applied algorithms, due to the great multithreading performance of most Android smartphones.

ISBN 978-80-86943-38-1



(a) iOS - Apple iPad Pro (b) Android - Samsung Galaxy S7

Figure 4: Successful loop closure

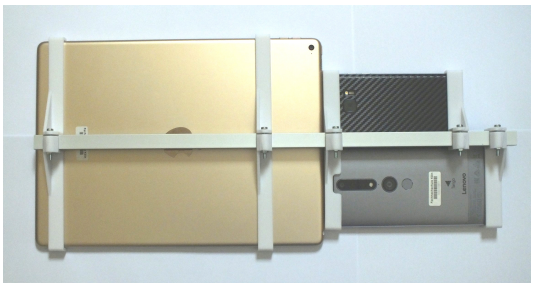


Figure 5: Test setup for fixing the devices. For testing purposes we added a Google Tango device, a now abandoned AR-project by Google that uses specialized hardware for tracking movement and the environment.

5.2 Experimental Performance

This section presents the results of the experimental comparison between the Android port and the iOS version. The two platforms are very different in some regards, especially in the sensor inputs, which have a direct impact on the quality and performance of the framework. Therefore, it makes little sense to use a publicly available dataset for the comparison, such as MH_03_medium used in [LQH*17] or a similar one. Instead, a test under practical conditions was chosen, exposing the different platforms to the same environmental conditions. Of course, physically it is impossible to produce the exact same conditions, since the cameras cannot all be at the same position. In order to provide the best possible approximation, a test apparatus was designed on which the devices can be firmly fixed. A picture of this setup can be seen in Figure 5.

Due to the lack of a possibility to acquire the ground truth of the traveled route, the result could only be evaluated manually. For that purpose the screen of each device was recorded. Since it is not possible to record a video of the screen on iOS version 10, we took screenshots at key points of the route on the iPad.

For the test, as the starting point we chose a feature rich object, which stood out from the rather repetitive surroundings. We then took a walk inside our university building. The covered distance is approximately 80 m. At the end of the test run we returned to the starting point, so that the algorithm had a chance to detect and close the occurring loop.

Both the iOS version and the Android port are able to recognize the previously visited location and adjust their pose accordingly, which can be subject to a significant drift, caused e.g. by barren corridors or fast movements. The screenshots in Figure 4 visualize this loop closure. As can be seen both show roughly the same quality in tracking. To enable measuring the occurring error, an additional online or offline system for recording the groundtruth would have to be installed. Furthermore the framework would need to be modified in a way that allows logging the estimated current pose over the course of the testing. After testing, the error could then be calculated and visualized.

6 CONCLUSION AND FUTURE WORK

We describe the process of porting a visual inertial SLAM algorithm from iOS to Android and discuss how to solve the according challenges. At the example of the VINS Mobile Framework we explain the necessary steps to calibrate the device specific parameters correctly and analyze its performance both quantitatively and qualitatively.

In future work, the complex calibration process could be partially automated, so that a broader range of android smartphones could be used by the framework. To further expand the functionality of the framework, techniques of dense environment reconstruction could be used to enable occlusion of AR-content by the environment.

Our framework is publicly available at [GIT].

ISBN 978-80-86943-38-1

7 REFERENCES

[ARC] ARCore - Google. URL: https://developers.google.com/ar [cited 29.04.2019].

[ARK] ARKit - Apple. URL: https://developer.apple.com/arkit [cited 29.04.2019].

[ARTa] ARToolKit. URL: https://github.com/artoolkit [cited 29.04.2019].

[ARTb] artoolkitX. URL: http://www.artoolkitx.org [cited 29.04.2019].

[BOHS15] BLOESCH M., OMARI S., HUTTER M., SIEGWART R.: Robust visual inertial odometry using a direct ekf-based approach. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Sep. 2015), pp. 298–304. doi:10.1109/IROS.2015.7353389.

[GIT] VINS-Mobile-Android. URL: https://github.com/jannismoeller/VINS-Mobile-Android [cited 27.04.2019].

[GSMa] GSMARENA.COM: Apple iPad Pro 12.9 (2015) - Full tablet specifications. URL: https://gsmarena.com/apple_ipad_pro_12_9_(2015)-7562.php [cited 29.04.2019].

[GSMb] GSMARENA.COM: Samsung Galaxy S7 - Full phone specifications. URL: https://gsmarena.com/samsung_galaxy_s7-7821.php#g930f [cited 29.04.2019].

[HKBR14] HESCH J. A., KOTTAS D. G., BOWMAN S. L., ROUMELIOTIS S. I.: Consistency analysis and improvement of vision-aided inertial navigation. *IEEE Transactions on Robotics 30*, 1 (Feb 2014), 158–176. doi:10.1109/TRO.2013.2277549.

[iFi16a] IFIXIT: iPhone 7 Plus Teardown, Sept. 2016. URL: https://de.ifixit.com/Teardown/iPhone+7+Plus+Teardown/67384#s136470 [cited 29.04.2019].

[iFi16b] IFIXIT: Samsung Galaxy S7 Teardown, Mar. 2016. URL: https://de.ifixit.com/Teardown/Samsung+Galaxy+S7+Teardown/56686#s122920 [cited 29.04.2019].

[IWKD13] INDELMAN V., WILLIAMS S., KAESS M., DELLAERT F.: Information fusion in navigation systems via factor graph based incremental smoothing. *Robotics and Autonomous Systems 61*, 8 (2013), 721 – 738. doi:10.1016/j.robot.2013.05.001.

[KUD] Kudan. URL: https://kudan.io [cited 29.04.2019].

[LLB*15] LEUTENEGGER S., LYNEN S., BOSSE M., SIEGWART R., FURGALE P.: Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research 34*, 3 (2015), 314–334. doi:10.1177/0278364914554813.

[LM13] LI M., MOURIKIS A. I.: High-precision, consistent EKF-based visual-inertial odometry. *The International Journal of Robotics Research 32*, 6 (2013), 690–711. doi:10.1177/0278364913481251.

[LQH*17] LI P., QIN T., HU B., ZHU F., SHEN S.: Monocular visual-inertial state estimation for mobile augmented reality. In *2017 IEEE International Symposium on Mixed and Augmented Reality* (Piscataway, NJ, 2017), Broll W., Regenbrecht H., Swan J. E., (Eds.), IEEE, pp. 11–21. doi:10.1109/ISMAR.2017.18.

[MAT] MatLab. URL: https://mathworks.com/products/matlab.html [cited 29.04.2019].

[MAX] MAXST AR SDK. URL: https://developer.maxst.com/ [cited 29.04.2019].

[OPE] OpenCV. URL: https://opencv.org [cited 29.04.2019].

[QLS17] QIN T., LI P., SHEN S.: VINS-Mono: A robust and versatile monocular visual-inertial state estimator, 2017. URL: http://arxiv.org/pdf/1708.03852.

[SMK15] SHEN S., MICHAEL N., KUMAR V.: Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (May 2015), pp. 5303–5310. doi:10.1109/ICRA.2015.7139939.

[VIN] VINS-Mobile - GitHub. URL: https://github.com/HKUST-Aerial-Robotics/VINS-Mobile [cited 29.04.2019].

[VUF] Vuforia. URL: https://engine.vuforia.com [cited 29.04.2019].

[WIK] Wikitude. URL: https://wikitude.com [cited 29.04.2019].

 ISBN 978-80-86943-38-1

ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

Open Data Visualization in Danish Schools: A Case Study

Mubashrah Saddiqa
Department of Electronic Systems
Aalborg University, Aalborg
Denmark
mus@es.aau.dk

Birger Larsen
Department of Communication
Aalborg University, Copenhagen
Denmark
birger@hum.aau.dk

Rikke Magnussen
Department of Communication
Aalborg University, Copenhagen
Denmark
rikkem@hum.aau.dk

Lise L. Rasmussen
Department of Communication
Aalborg University, Copenhagen
Denmark
liselykke@hum.aau.dk

Jens M. Pedersen
Department of Electronic Systems
Aalborg University, Aalborg
Denmark
jens@es.aau.dk

ABSTRACT

Increasingly public bodies and organizations are publishing Open Data for citizens to improve their quality of life and solving public problems. But having Open Data available is not enough. Public engagement is also important for successful Open Data initiatives. There is an increasing demand for strategies to actively involve the public exploiting Open Data, where not only the citizens but also school pupils and young people are able to explore, understand and extract useful information from the data, grasp the meaning of the information, and to visually represent findings. In this research paper, we investigate how we can equip our younger generation with the essential future skills using Open Data as part of their learning activities in public schools. We present the results of a survey among Danish school teachers and pupils. The survey focuses on how we can introduce Open Data visualizations in schools, and what are the possible benefits and challenges for pupils and teachers to use Open Data in their everyday teaching environment. We briefly review Copenhagen city's Open Data and existing open source software suitable for visualization, to study which open source software pupils can easily adapt to visualize Open Data and which data-sets teachers can relate to their teaching themes. Our study shows that introducing Open Data visualizations in schools make everyday teaching interesting and help improving pupils learning skills and that to actively use Open Data visualizations in schools, teachers and pupils need to boost their digital skills.

Keywords

Open data, visualization's tools, interactive visualizations, educational resource, pupils.

1 INTRODUCTION

Open Data is freely available data for anyone to use, share and re-publish it anywhere and for any purpose. Mostly, Open Data is published by governments, public sectors, researchers and organizations such as data about transport, budgets, business, environment, maps, science, products, education, sustainability, legislation, libraries, economics, culture, development, design and finance. Open Data has the power to revolutionize and disrupt the way societies are governed. Being used in different sectors on a wider scale, there are many examples of how Open Data can save lives and change the way we live and work [1]. Organizations in many countries are beginning to publish large data-sets which are open for the public. But, the availability of Open Data alone is not enough to ensure that it is made use and bring useful results. Without some form of aggregation, it can be hard for users to make sense of Open Data and understand it if they have no or little experience of processing and data analysis. Visualizations play an essential role as they are an effective way of interacting with large amounts of data from different fields, ranging from history [2] to economics [3] to basic science [4]. It provides a powerful means both to make sense of data and to present what has been discovered. With the advent of new interactive visualization techniques, presentation of data in a pictorial or graphical format makes it easier for a layman to understand large data-sets [5]. Presently, visualizations are widely used in news, books, internet, health, and economics [6] and having a limited knowledge of visualization can be a serious handicap.

As, society as a whole, is becoming increasingly digitized, it becomes essential to develop new research and educational models in schools to improve data-based digital competencies among students [7]. Many European countries have acknowledged the potential of digital competencies [8], and several have taken steps to introduce Open Data in schools and have started Open Data projects, e.g., Open Data for education competition in Northern Ireland with the intentions to use ex-

isting Open Data to assist teaching in primary and secondary schools [1].

The concept of Open Data can be made more interesting for school children and pupils by introducing visualizations and they may understand its potential and respond more quickly when exposed to visualization techniques. Therefore, to optimize the engagement of the young generation with Open Data, we believe that they must be familiarized with visualization tools, (e.g. charts, geographical maps and other types of representations) at an elementary level [9]. These simple visualizations become more powerful if they are created in relation to Open Data from domains that are of importance to the pupils, e.g. using data from their own city to detect and understand problems that are pertinent to their local area and everyday life [10]. With the availability of Open Data, new opportunities arise for all kinds of organizations, including government agencies, not-for-profits and businesses and allows them to come up with new ways of addressing problems in society. These include predictive health care [11], improving the transport system [12], [13] and transparency [14]. Open Data also opens up a wide variety of mostly unexploited possibilities of their use in education, e.g. in the form of visualizations. Although attention on visualization literacy [15], [16] continues to grow in different disciplines [17] there is still an ongoing discussion of the relative merits of different visualization platforms [18], [19]. Several studies [20], [21] illustrate people's limited knowledge in understanding data visualizations, indicating the urgency to address the problem. Recent studies [22], [23] pointed towards defining tactics to value visualization literacy, while others addressed how to improve it, using basic pedagogic ideas. Although many new teaching models [24], [25] have been presented with the purpose of increasing visualization literacy among the public, but not much attention paid on the ground level, i.e. schools. Open Data visualizations [26] based on local information can not only help to make visualization literacy easy and interesting but can also improve civic awareness and learning behaviors among pupils. Large and complex open data-sets can be further simplified and visualized for use in the teaching of, e.g. basic physics, mathematics and statistical methods, geography, social science and data handling in general. These data-sets not only make the teaching interesting and interactive but also develop skills among pupils to understand and ask questions about different facts of their local areas.

In this research work we reflect on the importance of Open Data visualizations on educational aspects, e.g. how Open Data visualizations can best facilitate education, what are the required skills to work with Open Data and its presentations at school level, and what would be the constraints and problems when working with Open Data visualizations at a basic school level? We address the following research questions:

- What would be the benefits of introducing Open Data and its visualization in the teaching tasks?
- How can Open Data visualizations be used in education especially at the school level?
- How can visualization facilitate understanding of Open Data in the educational domain?
- Which tools are considered user-friendly and effective in visualizing Open Data in schools?

We present results from a survey addressed to pupils and teachers to investigate these research questions. We briefly review the Open Data of the City of Copenhagen through its Open Data platforms. To visualize Open Data at a school level, we also analyzed user-friendliness of existing visualization technologies. We visualize concrete open data-sets of Copenhagen using existing visualization tools and presented the results to school pupils and teachers. These specific Open Data visualization examples are used during the interviews to provide a reference on how teachers could relate to them as part of their teaching. Their feedback is used to identify the interesting open data-sets, challenges and problems that need to be tackled in order to work with Open Data visualizations in the schools.

The paper is organized as follows: Section 2 describes the motivation and methodology. Section 3 presents the analysis of the open data-sets of the City of Copenhagen with some suggested educational open data-sets corresponding to specific educational domains and a short review of existing visualization techniques. Section 4, describes the setup and results of the pupils-teachers survey. The conclusion of the study is given in section 5.

2 MOTIVATION AND METHODOLOGY

There is an increased number of governments and companies using Open Data to offer new services and products to the citizen, which signaled Open Data as one of the building blocks for innovation. Using Open Data effectively could help to save hours of unnecessary waiting time on the roads and help to reduce energy consumption. Using Open Data, newly developed mobile applications aim to make our lives a little easier and by using these applications we could have access to real-time information to minimize travel time, e.g. MinRejseplan [2] mobile application in Denmark. To equip

[1] https://www.europeandataportal.eu/en/highlights/open-data-schools

[2] https://www.nordjyllandstrafikselskab.dk/Billetter---priser/MinRejseplan

 ISBN 978-80-86943-38-1

our younger generation with the required digital skills for future challenges, it is important that teachers are aware of and able to use Open Data as an educational resource that allows them to develop data and digital skills among younger generation of pupils.

Denmark is one of the leading countries with the most up-to-date Open Data. Denmark has a national platform [3] for the cities to publish Open Data, providing a common entry point for accessing Open Data. According to the Digital Economy and Society Index (DESI) 2018, Denmark is among the most digital countries in Europe[4]. The report documents that 94 percent of the country's public facilities and services are online and are highly advanced. Copenhagen, the capital of Denmark, is also famous for its smart city initiatives. The city has a large collection of Open Data available for its citizens. But, the social impact of Open Data is very limited due to the lack of public awareness according to Open Data Maturity report 2018 [27]. Therefore data literacy becomes imperative for the citizen of Denmark especially for the future generation, in order to make use and contribute actively for the improvement of digital services. There is a growing demand to take initiatives that aim at advancing digital skills and creating new interactive learning and teaching resources at the school level. Open Data is an open resource that can facilitates teaching with real information and allows pupils to develop data and digital skills. In this research work, we focus in particular on how to create understanding and representation of Open Data as an educational resource for public school pupils. We focus in particular on the city's many types of data and how to put them into use especially in an educational context. For example, there may be data about traffic, pollution, light and the use of different areas and facilities. These data-sets can easily relate to different subjects, such as Science, Mathematics or Geography subjects. At the same time, it is possible to compare this data with other areas of the city.

To understand how Open Data visualizations facilitate educational activities we conducted a survey with Danish public schools in Copenhagen. The survey is conducted from August 2018 to December 2018. Six Danish public schools, 10 school teachers and 21 school pupils of 7th grades aged between 13-14 years participated in the survey. The survey is formulated over interviews, questionnaire, and observations. The detailed methodology is discussed in Section 4 below. To investigate the research questions through our survey, we further sub-divided the research questions into three categories.

The Data Perspective:

- What would be the benefits of introducing Open Data in the teaching tasks?
- What types of data are already used in the schools?
- What would be the most interesting open data-sets?
- What benefits do teachers think Open Data visualizations could have?

Teachers/Pupils Competencies Perspective:

- Which skills and competencies do pupils and teachers already have in working with data and presenting them?
- Which skills and competencies would be necessary for pupils and teachers to work and present Open Data?

The Visualization's tool Perspective:

- Which visualization tools are appropriate and easy to adapt for school teachers and pupils?
- Which visualization tools do teachers already use in their teaching?
- What are the possible challenges in adopting new visualization tools, e.g. language barriers or installation of the systems?

To identify perspective on teachers/pupils skills and competencies, we collaborate with Danish public school teachers and pupils to learn their views about bringing Open Data into their classrooms and to identify skills they already have. The details are discussed in Section 4 below. To investigate perspectives about data and visualization tools, we reviewed and analyzed the open data-sets of the City of Copenhagen and identified which domains of Open Data could be used in the different domain of education, e.g. Mathematics, Science, and Geography. We also make a comparison of different available user-friendly visualization's tools to find out which of them could be the best option for school teacher and pupils or if there is a need to develop Open Data visualization interface specifically for schools.

In the next section, we will briefly review open data-sets of Copenhagen to discover possible data-sets which can be used as part of different subjects in schools and contains interesting information about the city and its local areas. And we analyze user-friendly visualization's tools to identify those we could adapt for the rest of the survey.

[3] http://www.opendata.dk/

[4] http://ec.europa.eu/information_society/newsroom/image/document/2018-20/dk-desi_2018-country-profile_eng_B43FFE87-A06F-13B2-F83FA1414BC85328_52220.pdf

ISBN 978-80-86943-38-1

3 ANALYSIS OF OPEN DATA-SETS AND USER-FRIENDLY TECHNOLOGIES

In this section, we first analyze the open data-sets of Copenhagen city to categorized Open Data themes corresponding to different educational domain, e.g. Science, Mathematics, Geography and Social science. In order to utilize the Open Data in education, we also focus on the availability of the data-sets, data formats and data types. Next for visualization of Open Data, we analyze existing user-friendly visualization's tools and discuss their main features.

3.1 Open Data-sets of Copenhagen

The Copenhagen Open Data website [5] has a large number of data-sets from all over the City. There are data-sets, for example on traffic, parking, the city's physical infrastructure, as well as data on population, culture and education. More than 280 data-sets are available in different formats and can be found in the Copenhagen city website. In order to make use of Open Data using visualizations at school level, it is important to understand the different formats and characteristics of data-sets. The Open Data of Copenhagen city can be categorized into ten general themes. These general themes of Open Data with corresponding sub-domains are listed in Table.1. The data sets includes static data, dynamic

General Themes	Sub-domain
Environment	Nature, Water Quality, Air Quality, Pollution, etc.
Governance	Demographics, Elections, Census, Transparency, etc.
Health & Care	Social care, Care Homes, Child Care, etc.
Infrastructure	Roads, Buildings, Locations, Planning, etc.
Transport	Traffic, Parking, Public Transport, Pedestrian, Cyclist, etc.
Community	Society, Housing, Employment, etc.
Education	Schools, Kindergartens, etc.
Energy	Solar Energy, Consumption, Carbon Emission, etc.
Culture & Sports	Entertainment, Tourism, Cultural Locations, etc.
Economy	Finance, Economy, etc.

Table 1: General Themes of Copenhagen Open Data

data, geographical and live data in different formats. In Table.2, the main characteristics of the Open Data of Copenhagen city are presented. Most of the data-sets are up-to-date and timely processed. The statistical contribution of major themes of Copenhagen city is

[5] https://data.kk.dk/

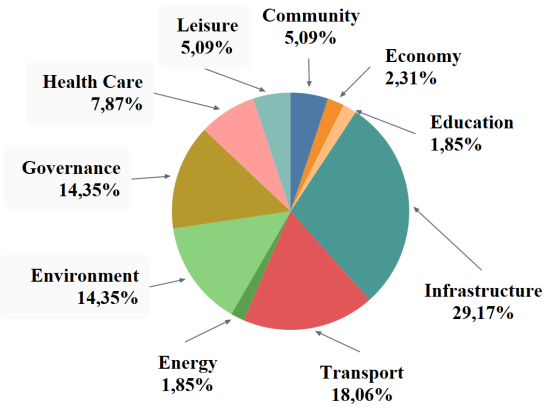


Figure 1: Statistical view of Open Data Major Themes

shown in Figure.1. This brief analysis will enable us to identify some of the relevant data categories which will be used as part of specific educational domains. The Open Data review of city enabled us to explore 4 impact domains (educational themes) discussed in [28] with associated sub-domains corresponding to main educational domains, i.e. Mathematics, Science and Geography. Figure.2, represents the possible mapping of

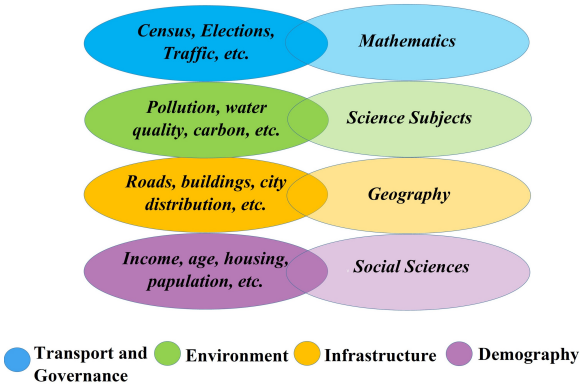


Figure 2: Example of Open Data themes corresponding to educational domains

Open Data educational themes to the educational sector. Table.3, presents different sub-domains of Open Data (educational themes) and examples of their possible use corresponding to specific educational domains. This analysis will enable us to ask teachers which of the different available data-sets and formats can be used during teaching tasks, which data-sets are interesting and how they could be used as an aid to make teaching tasks more interactive.

3.2 Review of User-Friendly Visualization Tools

A large number of data visualization technologies have been developed over the last decade to support the ex-

ISBN 978-80-86943-38-1

General Themes	Data Types	Format	Data-sets
Environment	Live/Sensors, Multi-dimensional	CSV, EXCEL	33
Governance	Statistical, Multi-dimensional	CSV, EXCEL, SHP(Shapefile)	22
Health and Care	Statistical	CSV, EXCEL	17
Infrastructure	Geographical, Live/Sensors, Statistical	CSV, EXCEL, DWG(Drawing)	63
Transport	Live/Sensors, Statistical	CSV, EXCEL, KLM, GeoJson	39
Community	Statistical, Historical	CSV, EXCEL	29
Education	Statistical	CSV, EXCEL, GeoJson	5
Energy	Live/Sensors, Statistical	CSV, EXCEL, KLM	4
Culture & Sports	Statistical, Historical	CSV, EXCEL	9
Economy	Statistical, Historical	CSV, EXCEL	9

Table 2: Characteristics of General Themes of Copenhagen Open Data

Educational Domain	Educational Themes	Examples
Sciences Subjects	**Environmental Data**: Pollution, water quality, traffic, carbon Level, energy, etc.	To view the city's pollution within the city, e.g. carbon level.
Mathematics	**Statistical Data**: Gender, population, age, housing, education, etc.	To make comparison of different details, e.g. gender comparison.
Geography	**Geographic Data**: City distribution, Buildings, Roads, Locations, etc.	To view the locations of buildings, roads and areas etc.
Social Science	**Demographic Data**: Education, nationality, income, work, culture, etc.	To view the details about population, age, income, employment and education etc.

Table 3: Open Data main themes and corresponding educational domain

ploration of large data-sets. In order to interact with Open Data, it is important to visualize it. For this, visualization technologies and software are required, which are user-friendly, needs no programming knowledge, and are supposed to be simple in use if used by the school teachers and pupils. According to The Tech Terms Computer Dictionary [6], user-friendly means a software interface that is easy to learn and easy to use. It should be simple with easy access to different tools and options, and with minimal explanation for how to use them. For school pupils, We are interested in tools which are

- user-friendly
- needs no coding/programming requirements and
- support the most common formats, (e.g. CSV, Excel, Google sheets etc.)
- provide a platform where teachers and pupils can share their visualizations
- provide a public forum for inspiration, collaboration with others and share experiences
- provide a free license for pupils and teachers

We analyzed six different tools which are often used for visualization purposes. In the next sections, we discussed their main features and compare them to one another based on the criteria defined above to find the best possible tools adapted for visualization at school levels. The comparison is given in Table.4.

Tableau

Tableau[7] is a data visualization tool and provides different products with a one-year free license under its academic program. The academic suite includes:

- Tableau Desktop
- Tableau Prep
- Tableau Online
- Tableau public to collaborate with others visualizations and experiences
- Customer support and community forum

Tableau Desktop is used to connect with various types of data and for creating visualizations. Tableau Prep transforms data for analysis and allows users to combine and clean data prior to visualization. With Tableau Online pupils can easily share and collaborate their work by uploading workbooks to a site managed by instructors. Sites are issued for 1 year and allow for 100

[6] https://techterms.com/definition/user-friendly

[7] https://www.tableau.com/

ISBN 978-80-86943-38-1

Tools	Free License	Academic Program	No Programming	Teachers' Site	User Friendly	Public Forum
Tableau	x	x	x	x	x	x
QlikView	x	x	x		x	x
Power BI	x		x		x	x
Excel	x		x		x	x
Datawrapper	x		x		x	x
Google Maps	x		x		x	x

Table 4: Comparison of commonly used open source software based on defined criteria

simultaneous users. There is no need for any coding knowledge to work with Tableau.

QlikView

QlikView[8] is a user-friendly, interactive open source visualization tool with no requirements of coding knowledge. Visualizations can be shared with up to 5 users. It is compatible with default and custom data connectors and also import data from popular databases. Qlik also provides an academic package for students and teachers which includes the following resources:

- Qlik Software
- Qlik Continuous Classroom online learning platform
- Data Analytic curriculum featuring lecture notes, on-demand videos, handouts, activities and real-world, interactive business use cases
- Qlik Community Academic Program Space, a forum for professors and students to access resources, collaborate with others and share experiences and Qlik Customer Support

Power BI

Power BI[9] is a collection of software services, apps, and connectors that work together to turn data into interactive visualizations with out any programming knowledge. Power BI easily connect to different data sources. Pupils can get a free limited version of Power BI but not an academic program with full resources. BI also provides a public forum to collaborate with others but it does not provide a separate free site manged by instructors to collaborate with their pupils.

Excel

Excel[10] is part of the well-known Microsoft Office suite and also supports data analysis. One of the greatest advantages of Excel is its flexibility, as it puts little or no constraints on the user's ability to create visualizations. Excel does not provides specific academic programs. No programming knowledge is required for visualization purposes. Excel also support commonly used formats, (CSV, Excel and sheets etc.). Excel provides low barriers and sufficient tooling, but it stops at a medium data level.

Datawrapper

Datawrapper[11] is a user friendly and mobile-friendly data visualization tool. No programming is required for visualizing data. It supports 10,000 monthly charts to publish. It is compatible with CSV, Excel and Google sheets and provides a community forum to share experiences and problems with others but there is no academic program for pupils or teachers.

Google Maps

We also list **Google Maps** as a visualization tool because geographical data can easily be visualized using Google Maps. In addition, it is a familiar tool for pupils and teachers and mostly are already familiar with many of its features, e.g. from planing travels, finding locations and receiving traffic information. It is also user-friendly and requires no programming skills.

We selected Tableau and Google Maps as the visualization tools, which we used as examples during the interviews. Tableau fulfills the required criteria and Google Maps is a good option as most of the people already familiar with its features and functions.

4 PUPILS-TEACHERS SURVEY

In order to identify the teacher's perspective on the role that Open Data visualization can play for school pupils, we surveyed Danish public school teachers and pupils. The survey includes interviews with teachers and a two-day pilot test with school pupils and teachers.

4.1 Participants

For teachers' study, we recruited 10 teachers from 6 different Danish public schools. The teachers were identified according to their subject, experience and age. The

[8] https://www.qlik.com/us/products/qlikview
[9] https://powerbi.microsoft.com/en-us/
[10] office.microsoft.com/excel
[11] https://www.datawrapper.de/

ISBN 978-80-86943-38-1

focus subjects were Mathematics, Science and Geography. The teachers have teaching experience from 3 years to 15 years in the above-mentioned subjects and were aged between 25-45 years. For pupils' study, we ran a pilot test for two days in one of the Danish public school. Two (science and maths) teachers along with 21 students of 7th grade, aged between 13-14 years old participate in the pilot test.

4.2 Setup

Before the interviews, we delivered presentations of about 10 to 15 minutes on Open Data and its visualization using data-sets containing local information pertaining to the neighborhood around each school, and their possible usage at the school level as part of Mathematics, Science, or Geography. For instance, data-sets about the areal distribution of the city can provide interesting information for a Geography class. We transform bigger data-sets within different themes into smaller data-sets to present the local information relevant to respective school locations using Tableau and Google Maps. For example, Figure.3, represents Copenhagen

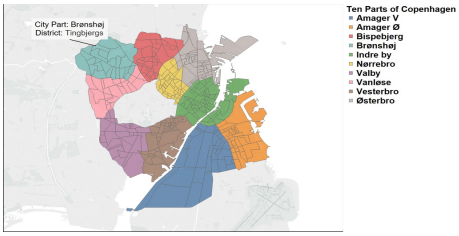


Figure 3: Areal distribution of Copenhagen

distribution into different parts, each with a unique color and further divided into smaller districts. It can be an interesting visualization for school pupils as they can locate their local areas easily. Figure.4 represents

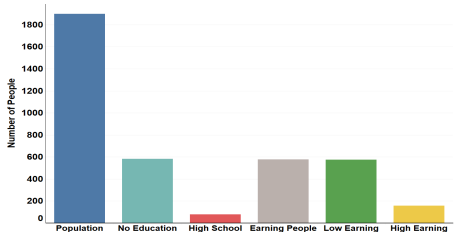


Figure 4: Statistical details of Tingbjerg

statistical details of Tingbjerg (district of Copenhagen) are presented, e.g population, education, low and high income. Using Google Maps, we visualize locations of the city's public schools according to their postal code numbers as a specific example for teachers and pupils. School pupils can use this map to get direction and measure the distance of their schools from their homes, different routes to their schools, more information about schools, e.g. schools website and address. These visualizations formulated over the local area information near the school, will help teachers to understand what type of data and information they could present to pupils during their teaching tasks as part of Mathematics, Science and other related subjects. For the pilot test with pupils, we requested the school administration to install the free academic package of Tableau on pupils and teachers computers. We provided teachers (math and science) beforehand with some specific visualizations which they could relate with their subjects, e.g. pollution level in different parts of the city at different times of the day, traffic passing through their areas at different parts of the day, the population details including age, education, and gender details. We also designed activities to explore the information through these specific visualizations, e.g. to find the carbon level of their area in morning and evening time, to find the gender details of their area etc. Pupils were also given a presentation, where we presented the visualizations of the data-sets from the educational Open Data themes given in Table.3, using Tableau.

4.3 Procedure

To investigate our research questions in different aspects discussed in Section 2, we conducted a teachers and pupils survey. The teachers' study was carried out with individual participants. Before the interviews, the participants were given a small introduction to Open Data and how they can adapt it as part of teaching tasks. We also asked both teachers and pupils for their consent to participate in the research work using video and audio devices. We used the Danish language for both teachers and pupils' study.

4.3.1 Teacher's Study

To achieve the best possible results and feedback on the three perspectives of our research questions, i.e.; the data perspective, the teachers perspective and the visualization tool perspective. We divide the teacher's study into three sessions, each session with 30 to 40 minutes. The overall time for teacher's study is 1.5 to 2 hours. To investigate the data perspective, in the first session, we presented short presentations of Open Data and its visualizations, existing open data-sets of Copenhagen according to the location of the schools and its possible usage at the school level as part of Mathematics, Science, and Geography subjects. To identify the visualization tools perspective, teachers will test the visualization tool Tableau, and try to visualize some of the provided ready to use open data-sets in the second session of the study. This will help us to identify which competencies teachers need when working with Open Data visualization and how convenient is it for the teachers to use. This is used as a background in order to investigate if they can use Tableau as part of their teaching. Teachers were provided with personal assistance to understand some main features of Tableau and Google Maps. In the third session, focus is given more on the

ISBN 978-80-86943-38-1

teachers' perspective. We asked participants to fill out a questionnaire and answer several semi-structured questions which elicit their views about the following perspectives.

Data perspective:

We asked questions about data already in use and formats used for the presentation of data during teaching especially in Mathematics, Science, and Geography tasks. What is the pupils' level of understanding data in different grades?

Teachers Perspective:

How Open Data visualization facilitate education in schools? What will be the possible impact on their teaching environment and on the school pupils with the use of real examples using Open Data visualization. What are the teachers' views about the skills pupils need to work with Open Data and its presentation?

Visualization tools Perspective:

We investigate the adapting of visualization tool Tableau and Google Map in order to visualize Open Data in a simple way. Whether they encountered any technical issues or limitations of the visualization tool? What will be the possible challenges and problems?

In addition, we asked several semi-structured and optional open-ended questions about the problems, and their suggestion to successfully introduce Open Data visualization into the educational domain.

4.3.2 Pilot Test

The role of pupils are important to investigate how they work with data, what data-sets are interesting for pupils about their areas and how they interpret Open Data visualization. We ran a pilot test in a Danish public school with 7th grade pupils aged between 13-14 years old. The pupils worked with Open Data in their Physics/Chemistry and Mathematics lectures for two consecutive days. We presented a 5-10 minutes presentation about Open Data with the help of visualizations of data-sets within educational themes. These include air pollution, carbon level, traffic around their area and some population details of their area. After the presentation, teachers relate some of the visualizations as part of Chemistry, where they talked about air pollution and different gases responsible for the pollution. With the help of Open Data visualization, teacher pointed different areas where pollution level is higher than the other areas, e.g. train and central bus stations. Similarly, Maths' teacher also relates statistical data of the city to make comparisons. On the second-day, pupils were taught how to use Tableau. They were provided with small data-sets to visualize. All these activities were observed and documented using videos and still photos. The pilot test ended with a questionnaire in the form of simple questions. We asked pupils about their understanding with different types of data formats, e.g. Excel, spreadsheets, CSV and the types of graphs, e.g. line, bar, pie charts. Which type of graph do they understand easily? What are the interesting data-sets about their city? Based on the input from teachers and pupils we also analyzed the three main issues.

4.4 Result

In this section, we discuss outcomes from the pupils-teachers survey. The following are the more salient points from their feedback.

Teachers/Pupils Perspective

- Teachers are able to understand and describe the Copenhagen Open Data visualizations presented during the interviews.
- According to teachers describing the city using Open Data visualizations will bring new perspectives in school teaching and school pupils.
- Teachers already used tools like GeoGabra, Excel and Google Maps for basic data handling and presentations.
- Pupil are already familiar with different types of graphical representations of data in Mathematics and Science subjects, e.g. lines, bar and pie charts.
- The pupils are likely to learn more when working with actual data of their area using visualizations.
- The interactive visualization tools can help a lot in order to make things more clear and interesting for pupils, but they are relatively hard to use.
- Pupils needs both, the skills and the tools to actively work with Open Data as an educational resource.
- Teachers believe that pupils' involvement can be made more interactive during the class using real data and real examples.

Data Perspective

- The data cleaning and preparing concepts are not often used in teaching.
- Pupils use mostly data in tables to make graphs. It is not seen as difficult to introduce other formats of data in teaching, e.g. CSV and Excel formats.
- Pupils can use statistical Open Data of Copenhagen to understand mathematical methods, e.g. to make the comparison of different real-life facts of their local areas.
- The teachers agreed that they can introduce different formats of Open Data particularly the CSV in 4th-7th grades and Excel format from 5th-7th grades in mathematics subject.

 ISBN 978-80-86943-38-1

Visualization's Tools Perspective

- Teachers are comfortable with Tableau and Google Maps and believe that pupils can take benefits from these open source tools.
- Pupils from grade 4th-7th are also used to Google Maps from previous teaching experience. It will give more meaning if they can make use of it in their daily life using Open Data, e.g. they can use Open Data to locate play areas on Google maps to get directions and measure distances from their schools or homes.
- The Open Data visualization can play an important role in explaining more abstract things in a concrete way.

4.5 Discussion

In order to fully enable school pupils to understand and work with Open Data visualizations, introduction to data formats, data preparation, and data cleaning concepts need to be introduced as part of their subjects for example in Mathematics. Using Open Data visualization of different data-sets, e.g. traffic, population, economy, and education etc., of their city in subjects like Mathematics, Geography and Science class can give them a chance to work with actual data. For example, presenting pollution level near the school at different periods of the day not only makes Science class interesting but also improve civic awareness among younger generation as the simple local visualization allow them to discuss why the level is higher at some periods and how they can reduce it. Using interesting statistics about pupils age group in different cities could help Mathematics in understanding ratio and percentage problems with real data. Teachers are comfortable with Tableau as visualization tool because of its quite easy to use functionally without any coding and free licensing for pupils and teachers but they also point out some limitations, e.g. they need extra time and efforts to visualize related open data-sets. Google Maps is also seen as a good option for geographical representations and measuring distances or finding alternative routes and can be used as part of Geography class. Based on the survey, we also identified some of the challenges. It can be hard for public school teachers to directly use the Open Data available at city's Open Data portal. They will need smaller ready-to-use data-sets as part of their teaching subjects. Teachers need support and training workshops in order to use new visualization technologies at schools. The visualization tool Tableau and other open source tools are in English which could be a problem for public school pupils, but explaining some of the frequent functions in Danish can solve this problem. In order to bring Open Data in schools as an educational resource, our survey also suggests the development of a school friendly Open Data visualization interface that provides local real information in the form of interesting and simple visualization in pupils' native language as part of different subjects.

5 CONCLUSION AND FUTURE WORK

In this paper, we have presented a pupils-teachers survey as a first step, to study how Open Data visualization can be introduced in the schools to facilitate educational activities, which skills school teachers and pupils already have and which visualization tools could be used to visualize Open Data as part of teaching subjects. Our study includes the visualization of Copenhagen open data-sets, a review of existing user-friendly visualization techniques and software and a qualitative survey of teachers and pupils. Based on our survey, we concluded that Open Data visualization can be brought into schools as part of different teaching subjects especially Mathematics, Science, and Geography and it can plays an important role in improving the pupils learning behaviors, as teachers and pupils are able to relate real problems from different perspectives in everyday teaching. In order to use Open Data Visualization actively in different teaching subjects, teachers need to boost their digital skills, adopt new interactive learning and teaching resources as well as support the approaches for knowledge development. Insight from this study, advances the knowledge of our community about what are the skills needed for schools teachers to use the Open Data visualization in their teaching areas and how the availability and literacy of Open Data visualization in schools create opportunities for school pupils to come up with new ways of addressing and understanding society's problems.

As a future work, we are aiming to develop a school-friendly Open Data visualization interface that provides aid to teachers and pupils to visualize local area data-sets within Open Data educational themes, compare these data-sets with other areas and suggests different activities for pupils as part of teaching tasks. We will first identify needs and requirements for school-friendly Open Data interface using the Requirements Engineering domain and then develop prototypes for testing and validation of the interface in the Danish public schools.

6 REFERENCES

[1] Dong, H., Singh, G., Attri, A., and El Saddik, A. Open data-set of seven Canadian cities. IEEE Access, Vol.5, pp.529-543, 2017.

[2] Friendly, M. A brief history of data visualization. In Handbook of data visualization, Springer, pp.15-56, 2008.

 ISBN 978-80-86943-38-1

[3] Schwabish, J.A. An economist's guide to visualizing data. Journal of Economic Perspectives, Vol.28, No.1, pp.209-34, 2014.

[4] Jipeng, L., Fang, Y., and Yu, H. The Application of visualization in scientific computing technology to architectural design. International Forum on Computer Science-Technology and Applications, Vol.3, pp.93-96, 2009.

[5] Valkanova, N., et al. Public visualization displays of citizen data: design, impact and implications. International Journal of Human-Computer Studies, Vol.81, pp.4-16, 2015.

[6] Segel, E. and Heer, J. Narrative visualization: Telling stories with data. IEEE transactions on visualization and computer graphics, Vol.16, No.6, pp.1139-1148, 2010.

[7] Geoghegan-Quinn, M. Science 2.0: Europe can lead the next scientific transformation. In EuroScience open forum (ESOF), Keynote speech, 2014.

[8] Wastiau, P., et al. The use of ict in education: a survey of schools in europe. European Journal of Education, Vol.48, No.1, pp.11-27, 2013.

[9] Alper, B., et al. Visualization literacy at elementary school. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, ACM, pp.5485-5497, 2017.

[10] Atenas, J., and Havemann, L. Open data as open educational resources: case studies of emerging practice. Open Knowledge-Open Education Working Group, 2015.

[11] Ghassemi, M., Celi, L.A., and Stone, D.J. State of the art review: the data revolution in critical care. Critical Care, Vol.19, No.1, doi: http://ccforum.com/content/19/1/118, 2015.

[12] Dawes, S.S., Vidiasova, L., and Parkhimovich, O. Planning and designing open government data programs: An ecosystem approach. Government Information Quarterly, Vol.33, No.1, pp.15-27, 2016.

[13] Urry, J. Mobilities: new perspectives on transport and society. Routledge, 2016.

[14] Carlo Bertot, J., Jaeger, P.T., and Grimes, J.M. Promoting transparency and accountability through ICTs, social media, and collaborative e-government. Transforming government: people, process and policy, vol.6, No.1, pp.78-91, 2012.

[15] Abilock, D. Visual information literacy: Reading a documentary photograph. Knowledge Quest, Vol.36, No.3, pp.7-14, 2008.

[16] Taylor, C. New kinds of literacy, and the world of visual information. Literacy, 2003.

[17] Boy, J., Rensink, R.A., Bertini, E., and Fekete, J.D. A principled way of assessing visualization literacy. IEEE transactions on visualization and computer graphics, Vol.20, No.2, pp.1963-1972, 2014.

[18] Kim, S-H., Boy, J., Lee, S., Yi, J.S. and Elmqvist, N. Towards an Open Visualization Literacy Testing Platform. IEEEVIS Workshop, 2014.

[19] Wolff, A., et al. Data literacy for learning analytics. In Proceedings of the Sixth International Conference on Learning Analytics and Knowledge, pp.500-501, ACM, 2016.

[20] Börner, K., Maltese, A., Balliet, R.N., and Heimlich, J. Investigating aspects of data visualization literacy using 20 information visualizations and 273 science museum visitors. Information Visualization, Vol.15, No.3, pp.198-213, 2016.

[21] Kennedy, H., Hill, R.L., Allen, W., and Kirk, A. Engaging with (big) data visualizations: Factors that affect engagement and resulting new definitions of effectiveness. First Monday, Vol.21, No.11, 2016.

[22] Boy, J., Rensink, R.A., Bertini, E., and Fekete, J.D. A principled way of assessing visualization literacy. IEEE transactions on visualization and computer graphics, Vol.20, No.12, pp.1963-1972, 2014.

[23] Lee, S., Kim, S.H., and Kwon, B.C. Vlat: Development of a visualization literacy assessment test. IEEE transactions on visualization and computer graphics, Vol.23, No.1, pp.551-560, 2017.

[24] Huron, S., Carpendale, S., Thudt, A., Tang, A., and Mauerer, M. Constructive visualization. In Proceedings of the 2014 Conference on Designing Interactive Systems, pp.433-442, ACM, 2014.

[25] Ruchikachorn, P., and Mueller, K. Learning visualizations by analogy: Promoting visual literacy through visualization morphing. IEEE transactions on visualization and computer graphics, Vol.21, No.9, pp.1028-1044, 2015.

[26] Graves, A., and Hendler, J. Visualization tools for open government data. In Proceedings of the 14th Annual International Conference on Digital Government Research, ACM, pp.136-145, 2013.

[27] Open Data Maturity in Europe 2018: New horizons for Open Data driven transformation. European data portal, 2018.

[28] Saddiqa, M., et al. Bringing Open Data into Danish schools and its potential impact on pupils. 15th International Symposium on Open Collaboration, ACM, 2019.

 ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

Prototyping a Game Engine Architecture as a Multi-Agent System

Carlos Marin
Institute of New Imaging Technologies
Universitat Jaume I
Spain, Castellón
cmarin@uji.es

Miguel Chover
Institute of New Imaging Technologies
Universitat Jaume I
Spain, Castellón
chover@uji.es

Jose M. Sotoca
Institute of New Imaging Technologies
Universitat Jaume I
Spain, Castellón
sotoca@uji.es

ABSTRACT

The game engines are one of the essential and daily used applications on the game development field. These applications are designed to assist in the creation of this type of contents. Nevertheless, their usage and development are very complex. Moreover, there are not many research papers about the game engine architecture definition and specification. In this sense, this work presents a methodology to specify a game engine defined as a multi-agent system. In such a way, from a multi-agent approximation, a game engine architecture can be prototyped in a fast way. Also, this implementation can be exported to a general programming language for maximum performance, facilitating the definition and comprehension of the mechanisms of the game engine.

Keywords

Game engine architecture, game development, multi-agent system.

1 INTRODUCTION

The game engines are frameworks composed by tools and interfaces that increase the abstraction level over low-level tasks to develop a video game [Gre14]. They are designed to simplify the creation of video games, encouraging the reuse of components and by abstracting the communication with the hardware and the operating system running the game [Nys14]. This method reduces development times since it provides tools to solve common issues on most of the games. Each game engine has its own components organisation structure known as architecture. This architecture determines the organisation of the modules composing the engine and the communications between them, the operating system and the hardware drivers.

Even though its goal is to ease game development, these applications are not easy to use. Game engine development is a complex task despite the reduced number of academic papers on game engine architecture [And08, Amp10]. The current literature deals with the engine components, such as the behaviour specification, the scene render or the networking. Nevertheless, the game engine architecture connecting all these is a subject that has been barely covered. In fact, [And08] brings forward the lack of academic papers and research lines on this matter. Some of these lines are the identification of the software components common to every kind of game, and the research of common elements to every kind of game engine in order to define a genre-independent reference architecture and the recognition of the best practices on game engine development.

Multi-agent systems (MAS) are adapted to the distributed problem resolution, where multiple modules work autonomously over a shared environment to achieve a goal. In this sense, the primary objective of this work is to demonstrate how MAS are capable to easily specify the system's components that define a game engine architecture. For this purpose, a game engine architecture prototype based on MAS has been designed, with autonomous components, so that they can perform specific tasks on the elements of the game. The proposed game engine must be able to run completely functional games from this agent-based architecture. To this end, the prototype has been developed on the MAS development platform *NetLogo* [Tis04].

The remainder of the work is organised as it follows: state of the art on game engine architecture and its relationship with MAS is presented in section two. Section three shows a summary of the principal MAS features. Subsequently, the game engine architecture proposed on this work is presented in section four. Finally, in section five, the results are shown and subsequently discussed in section six, and in section seven, the conclusions of this work are presented along with the future research lines.

2 STATE OF THE ART

This section presents a compilation of technical works that associate MAS with the game engine and

their appearance on academic papers. Game engine architecture is defined as the organisation structure of the engine's components and their relationship with its supporting drivers, hardware and operating system [Gre14]. In a game engine, the components are the subsystems responsible for running specific tasks such as the rendering, the logic evaluation, the user interaction or the physics evaluation. Some of these subsystems can be considered as complete engines by themselves due to its complexity. In fact, the logic component is not easy to standardise due to its inherent connection with the mechanics of each game [Mil09, Lew02, Doh03, Amp10].

Among all the methods trying to specify the game engine components, the MAS is paradigmatic because it relies upon autonomous entities communicating with one another and performing tasks on a shared environment, and so it can easily relate the game and the behaviour specification elements [Nys14]. This cooperative distributed problem-solving fits perfectly with the task distribution on a game engine. Furthermore, the academic papers linking games and MAS becomes evident the implicit connection between these interactive systems and the agent-based systems in the following aspects: the game element's and the agent's concepts and their interrelation, their communication protocols and their cooperation mechanisms [Woo02, Gla06, Pos07, Sil03].

In [Pon13], the authors propose an agent-based system to control the game parameters according to the game objectives. Besides, [Fin08] shows a system where the agents learn autonomously to play games without human intervention, [Gra13] proposes a system to integrate virtual worlds with a multi-agent interface and [Jep10] proposes an agent creation framework for serious games.

Furthermore, [Dig09] shows the relationship between MAS and the game mechanics design, emphasising on the industry tendency to associate the game mechanics definition and the natural language on the game development. Moreover, [Bec14] develops a MAS based on the commercial game engine Unity, by doing a 3D simulation of the behaviours related to the path-finding methods for multiple agents on a passenger airport context.

Additionally, on the multiuser issue, [Ado01] presents a MAS to handle the multiuser mechanics on a tournament game, [Sac11] presents an intelligent agent-based distribution to build multiplayer systems and [Ara08, Ara12] introduces a MAS to conduct the design of Multiplayer Massive Online Games (MMOG), a specific type of games where the priority is the real-time interactivity of several game agents.

Lastly, [Gar06, Gar07, Gar10] shows a virtual environment where the agents are communicating with the player like in a 3D chat and then [Rem15] introduces an application specification for a 3D virtual fair as a MAS.

3 MAS FEATURES

According to the M. Wooldridge definition [Woo02], an agent is a computer system that is situated in an environment in order to meet its design objectives. Based on the general definition for a MAS, it is necessary to consider the following formal characteristics:

- The environment of the MAS can be in any of the discrete states of a finite set $E = [\ e_0, e_1, e_2, ...]$ of states.
- The agents in the system have a set of available actions with the capacity to transform their environment $Ac = [\ \alpha_0\ , \alpha_1\ , \alpha_2\ , ...]$.
- The run r of an agent on its environment is the interlayered sequence of actions and environment states r: $e_0 \rightarrow^{\alpha_0} e_1 \rightarrow^{\alpha_1} e_2 \rightarrow^{\alpha_2} \ldots e_{u-1} \rightarrow^{\alpha_{u-1}} e_u$, where $R = [r,\ r',\ ...]$ represents the set of all the possible and finite sequences of E and Ac.
- The effect of the agent actions on an environment comes determined by the transformation function τ: $R^{AC} \rightarrow \wp(E)$ [Fag95], where R^{AC} represents the subset of R ending on an action, and R^E represents the subset of R ending on an environment state.

In this sense, the following definitions can be established:

Definition 1: An environment *Env* is defined as a triple $Env = <E, e_0, \tau>$, where E is a set of states, $e_0 \in E$ is the initial state and τ is the state transformation function.

Definition 2: An agent *Ag* is defined as a function Ag: $R^E \rightarrow Ac$ establishing a correspondence between runs and actions. It is assumed that these runs end on an environment state [Rus95].

In this sense, an agent selects the action to perform based on the system's history that it has witnessed. It is necessary to take into account that the environments are implicitly non-deterministic, but the agents are deterministic. The set of all the agents *Ag* of a system is represented as *AG*. The set of all the runs of an agent *Ag* on an environment *Env* is *R*(*Ag*, *Env*).

Definition 3: A purely reactive agent [Gen87] is defined as a function Ag: $E \rightarrow Ac$, which indicates that their decision-making process only with the current environment state.

Definition 4: An agent is considered as a perceptive agent when it is composed of a perception function and an action function. It is represented as $Ag = <see, action>$, where *see* is a function *see*: $E \rightarrow Per$ and *action* is a function *action*: $Per^* \rightarrow Ac$.

 ISBN 978-80-86943-38-1

4 ARCHITECTURE PROTOTYPE

In order to define the game engine architecture, this study is focused on the engine's components definition and the behaviour specification for the game elements. In this sense, the formal correspondences with a MAS are established with the aim of proving that the agent-based engine is capable of making games. On this architecture, the MAS is the element that structures the information distribution and communication between the engine's elements. The prototype design is oriented towards the creation of 2D games, and it has been developed on the agent-based programming language *NetLogo* [Tis04].

Next, the elements of the game engine are presented from a MAS point of view:

- The engine environment, or the space of coexistence for every element of the game.
- The engine's agents, the elements equivalent to the engine components.
- The game's agents, also known as game objects or actors.

4.1 The engine environment

In this game engine, the environment *Env* represents the conjunction by the agents representing the engine's components or engine's agents and the game objects, also known as actors. This environment *Env* is in charge of storing the game's states E through its properties. Furthermore, starting from an initial state e_0, the modules perform transformations τ on the general properties of the environment and on the actors with which they have dependencies. The initial state e_0 determines the set of original properties present in the environment. Among the properties that define the state of the engine, there are the following categories:

- **Basic properties**: Properties related to the resolution of the output screen, the characteristics of the camera and the sound of the game.
- **Physic properties**: Characteristics that control the physical behaviour of the environment such as gravity or air friction.
- **Timers**: Set of properties responsible for controlling and providing information to the game elements on execution times, time differentials and elapsed time.
- **Lists**: The lists are responsible for maintaining the lists of actors used by each engine component, for storing the input events that have occurred in the engine and to manage the sounds queue.

The execution of the environment on the engine's architecture allows evaluating the processes related to the correct functioning of the game. The execution of the game $R(AG, Env)$, represents the application of the set AG for all agents of the engine respect to the environment *Env*. The process that relates the agents to the elements of the game is known as the game loop: the cycle of evaluation, updating and continuous execution of the game. It is responsible for managing the interaction between the engine's agents and actors; that is, it is the process through which the agents that represent the engine components apply their behaviour rules on the environment.

The engine agents' communication with the environment is not the same for all elements that compose it. Also, this engine agents work asynchronously and at different frequencies, however its sequential implementation is absolutely valid. Figure 1 shows a diagram describing the communications between the engine environment, the engine agents, and the actors, where the processes of information transmission are determined.

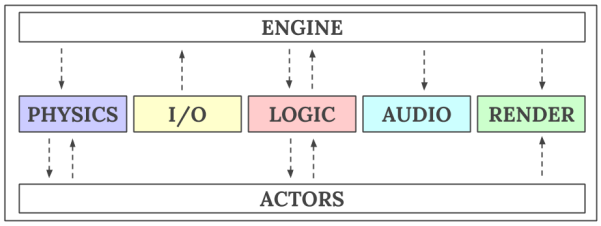


Figure 1: Communications' diagram between the components of the game engine.

4.2 The engine's agents

The engine agents that perform the role of the game engine components are the *Physics agent*, the *Input agent*, the *Logic agent*, the *Audio agent* and the *Render agent* (see Figure 1). These agents represent autonomous components that execute actions α on the environment *Env*. An agent *Ag* is an element that acts from the environment state e and its internal state, including the tasks assigned to it. In the designed architecture, engine agents represent the components of the game engine. Their behaviour is determined by predefined behaviour rules that evaluate the environment properties and the game actors, and which determine the actions to be taken. These agents are responsible for executing the actions *Ac* on the environment *Env*.

The engine agents present a reactive behaviour since they carry out their decision-making processes taking into account the present state of the environment and the environment's history that they have witnessed. Moreover, the *Input agent* and the *Audio agent* present a perceptual behaviour, and some of their essential functions are to wait until the perception function registers events that they can interpret. The different game engine agents are described below:

ISBN 978-80-86943-38-1

- **Physics agent**: Responsible for evaluating the actor's physical behaviour according to the environment properties and its configuration. The execution of this agent requires a high-frequency cyclic performance due to the need to iterate the numerical integration of the equations repeatedly. Also, this agent executes actions on the environment in charge of detecting collisions, applying the response to the collision, integrating the movement equations and updating the position of the actors.

- **Input agent**: Manages the communication between the engine and the user. It transforms input events into interpretable information for the engine. It has an asynchronous behaviour that stores its input data into the event list of the engine.

- **Logic agent**: Observes the environment state and runs the actor's behaviours rules so that they fulfil their tasks. It is a cyclic operation, with lower latency than the physics agent one.

- **Audio agent**: Reproduces sounds asynchronously while the sound queue is not empty.

- **Render agent**: Draws the environment state on the screen in a repeatedly with sufficient latency to meet the threshold of human vision.

The communication between engine agents
Next, a brief description of how the communication is established between the agents of the engine with the environment and the game actors:

- The *Physics agent* must access the properties of the actors to apply the transformations resulting from the evaluation of their physical behaviour. For these evaluations, it is relying on the physical set-up of the game, stored within its properties.

- The *Input agent* obtains the information from the events that are given from the operating system. It notifies the engine, with the aim that other engine agents, such as the *Logic agent*, can access this information.

- The *Logic agent* is the only agent with the ability to read and write both in the environment properties and in the general properties of the actors. This feature is necessary in order to set and to get all the necessary information for any required game mechanics.

- The *Audio agent* works on its reproduction queue. This queue is loaded by the *Logic agent* when it is required to play a sound.

- The *Render agent*, for its correct operation, requires information about the environment with the resolution of the screen or the properties of the camera as well as the actors, with their textures, dimensions and transformations.

4.3 The game's agents

The game actors are the agents that make up the games: characters, scenarios and props; and are in charge of performing the game mechanics. All of them have common characteristics and include a set of properties that determine their appearance and behaviour. The actors have the following types of properties:

- **Basic properties**: These properties are related to the position, rotation and scale of the actors.

- **Rendering properties**: The properties connected to the image representing the game actor and its visualisation. Also, it can display text defined by a font, a size, a colour and a style property.

- **Physics properties**: Properties that allow defining the physical characteristics of the actors such as speed, angular velocity, material properties (density, friction, restitution).

- **New properties**: Also, the actors can define new properties that expand their property list.

The *Logic agent* controls the actor's behaviour. Each game mechanic is determined by a set of behaviour rules stored in the actors. These behaviour rules are defined by logic statements, which in this case, are taking the form of a script. In the *NetLogo* case, they are implemented as functions.

5 THE USE CASE AND RESULTS

After the definition of the architecture, a use case is presented below to test the capabilities of the engine. The implementation is based on the classic arcade game *Arkanoid*, where the user must manage a paddle to control the bounces of the ball and destroy the most significant number of bricks without letting the ball colliding with the lower limit of the screen. The selection of this game as a demonstrator is because it includes features that require all engine components. The implementation of this example requires engine agents controlling the execution of the game and actor agents to perform the mechanics of the game. Nevertheless, the following behaviour algorithms are adapted to the case of use in order to simplify the presentation. Additionally, as stated above, there is only one type of actor agent: all the *Arkanoid* game elements are actors with configured properties and behaviour rules to fulfil their specific tasks.

 ISBN 978-80-86943-38-1

For the game development, a *Paddle actor*, a *Ball actor* and four sets of fourteen *Brick actors* have been created. Also, there is an additional actor whose task is only to store and to show the game score as a new property. Initially, the *Brick actors* are distributed in a grid at the top of the screen, the *Ball actor* is initialised over the *Paddle* with a constant speed on the y-coordinate and a random velocity on the x-coordinate, and the *Paddle actor* is centred on the bottom. A capture of this layout is represented in Figure 2.

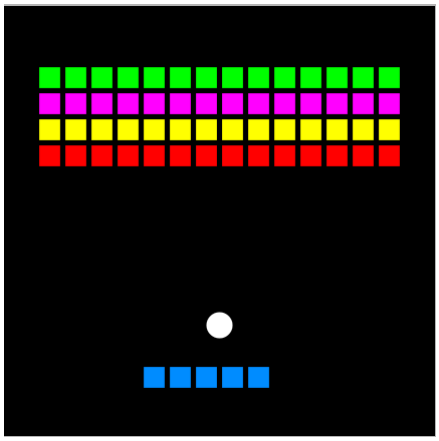


Figure 2: Capture of the *Arkanoid* game implemented with the prototyped engine.

Algorithm 1: Declaration of the environment properties, engine agents and actors agents in *NetLogo*.

```
globals [delta-time previous-time current-time]
physic-own [physic-list gravity-x gravity-y
  air-friction ]
input-own [event-list]
logic-own [logic-list]
audio-own [sound-list]
render-own [render-list resolution-x resolution-y
  camera-x camera-y camera-rotation camera-zoom]
actors-own [velocity-x velocity-y restitution
  active-physics? active-logic? active-audio?
  active-render? static?]
```

The declaration of these properties is presented in Algorithm 1, along with the engine agents properties and the actor agent properties. This property set can accept new properties depending on the game requirements. Moreover, during the game cycle execution, each engine agent evaluates the state of the environment and executes its actions according to the tasks it has programmed (see Algorithm 2).

The first step on the game loop is carried out by the *Physics agent* (see Algorithm 3). The execution of its functions begins with the *collision-detection* function, which is responsible for detecting collisions between

Algorithm 2: The game engine's game loop.

```
to go
  physic [ run-physic ]
  input [ run-input ]
  logic [ run-logic ]
  audio [ run-audio ]
  render [ run-render ]
end
```

actors. Next, the *collision-response* function is responsible for resolving the collision between two actors, which applies only to the actors with the active-physics property active. If any of these actors have a static physical behaviour, the response to the collision will not affect them. Finally, the *motion-integration* function is in charge of applying the motion equations to the actors. These movements are determined by the initial speed settings, the effect of gravity, friction with the air and the results of collisions. Its scope is about the actors with physical properties, as long as they are not static.

Algorithm 3: Physics agent behaviour loop.

```
to collision-detection
  ask actors [
    if (distance ("ballID")) < size [ set colliding true
  ]
  ]
end
to collision-response
  ask actors [
    if (colliding and active-physics? and (not
  static?)) [
      set velocity-y (velocity-y * -1 * restitution) ]
    ]
  ]
end
to motion-integration
  ask actors [
    if (active-physics? and (not static?)) [
      set velocity-x (velocity-x + gravity-x *
  delta-time)
      set velocity-y (velocity-y + gravity-y *
  delta-time)
      set xcor (xcor + velocity-x)
      set ycor (ycor + velocity-y)
    ]
  ]
end
```


ISBN 978-80-86943-38-1

Upon the *Physic agent* completion, the *Input agent* behaviour for this use case is presented in Algorithm 4. This behaviour is based on the *event-management* function, the function responsible for managing the procedures that determine if an input event has occurred in the engine. When an occurrence is detected, the list of events of the environment properties is modified to communicate that information to the rest of the environmental elements.

Algorithm 4: Input agent behaviour loop.

```
to event-management
  if left-key [ set left-key-event true ]
  if right-key [ set right-key-event true ]
end
```

Next, the environment is evaluated by the *Logic agent*. In the general case, this procedure is responsible for evaluating the logic of each actor agent in terms of game mechanics. For this use case, two rules are presented to demonstrate the running of its behaviour. Algorithm 5 shows two rules based on the function *evaluate-actors*.

The first function controls the behaviour of the *Brick actor*. After the detection of a collision event involving the *Ball actor* and a *Brick actor*, the *Ball actor* gives the response to the collision to simulate the rebound, and the *Brick actor* in question is eliminated from the game. Additionally, the property points of the *Score actor* is increased by one, and a sound is added to the playback queue. Conversely, the second function sets the movement of the *Paddle actor* after the user inputs. Keyboard events control the movement of the *Paddle actor*: it applies a shift to the left and a shift to the right with the *A* and *Z* keys respectively. This displacement will occur as long as it does not reach the limits of the screen.

After adding a sound to the playlist, the *Audio agent* is responsible for its reproduction. The *Audio agent* behaviour cycle is presented in Algorithm 6, where the *play-sounds* function sets the audio playback based on the reproduction list information.

Finally, the last step on the game loop is performed by the *Render agent*. This agent executes the procedures of drawing the scene on the group of visible actors. Nevertheless, the rendering functions in *NetLogo* are minimal. For this example, the function *draw-actors* is composed just by a refresh call with the native function *tick*.

On a separate issue, the end of the game is reached when the *Ball actor* has no more *Brick actors* left to destroy or when he reaches the lower limit of the screen, that is when the player handling the *Paddle actor* is not able to return the *Ball actor* to the *Bricks* zone.

Algorithm 5: Logic agent behaviour loop.

```
; behaviour rule for the Brick actor
to evaluate-actors
  ask actors [
    ifelse (static? and colliding) [ die ] [ set points
 points + 1 ]
  ]
end
; behaviour rule for the Paddle actor
to evaluate-actors
  if left-key-pressed [
    ask (actor "paddleID") [
      if xcor > min-pxcor + size [
        ask actor "paddleID" [ set xcor xcor - size ]
      ]
    ]
  ]
  if right-key-pressed [
    ask (actor "paddleID") [
      if xcor < max-pxcor - size [
        ask actor "paddleID" [ set xcor xcor + size ]
      ]
    ]
  ]
end
```

Algorithm 6: Audio agent behaviour loop.

```
to play-sound
  ask actors [
    if empty? sound-list [
      sound:play-note "Gunshot" 60 64 0.25
    ]
  ]
end
```

It can be seen that the programming language for MAS *NetLogo* is not prepared to make aesthetically attractive games. For example, it is not possible to add new images or new sounds to agents; it only works with predefined elements. Also, it is not possible to scale these forms in a single direction, which significantly limits the possibilities to the game aesthetics. In any case, it easily allows creating a game engine prototype that can be implemented on a more suitable programming language.

ISBN 978-80-86943-38-1

6 CONCLUSIONS AND FUTURE WORK

The design of the game engine architecture presents a methodology to specify a game engine defined as a MAS but also has some characteristics that are interesting to analyse. In the first place, it should be noted that the autonomous behaviour of the engine components over a shared space makes possible the resolution of many problems inherent in the development of games. Since each of them has autonomous tasks that could be done without external intervention. Besides, the engine is enriched with the essential characteristics to create a wide variety of games, where some fancy features are no longer necessary for the conventional 2D game engine.

Conversely, it should be noted that games are played with a single type of actor agent and, there are no hierarchical relationships between them. All the elements that define a game: the markers, the player, the NPC; they have the same properties and the same behaviour rules system. Furthermore, it is not necessary to define a scene graph, which simplifies the internal architecture of the engine and the design of the games. There are no complex data structures such as vectors or matrices that are not necessary for the creation of most arcade games.

In summary, the objective of this work aims to define a simple architecture prototype that is capable of running a game. It starts from the formal definition of the MAS with the end of leading a definition of its essential elements. Besides, the study of game engines and their relationship with MAS has allowed to generate a broad knowledge about the architecture of commercial game engines and to establish a game engine specification intuitively and closer to the way of describing systems with natural language. The engine is the environment of the MAS, while the components are the agents of the engine and the actors are the agents of the game. From the environment state, agents can perceive information and react to certain states based on their predefined tasks. The behaviour associated with the tasks is determined by behaviour rules and a pre-established set of actions.

About the future works, this prototype has led to the creation of a game engine following the architecture designed on this work. It is being built over the programming language *JavaScript* in order to execute games on web browsers.

7 ACKNOWLEDGMENTS

This work has been supported by the Ministry of Science and Technology (TIN2016- 75866-C3-1-R) and the Universitat Jaume I research project (UJI-B2018-56).

8 REFERENCES

[Gre14] Gregory, J. (2014). Game engine architecture. AK Peters/CRC Press.

[Nys14] Nystrom, R. (2014). Game programming patterns. Genever Benning.

[And08] Anderson, E. F., Engel, S., McLoughlin, L., Comninos, P. (2008). The case for research in game engine architecture.

[Amp10] Ampatzoglou, A., Stamelos, I. (2010). Software engineering research for computer games: A systematic review. Information and Software Technology, 52(9), 888-901.

[Tis04] Tisue, S., Wilensky, U. (2004, May). Netlogo: A simple environment for modeling complexity. In International conference on complex systems (Vol. 21, pp. 16-21).

[Mil09] Millington, I., Funge, J. (2009). Artificial intelligence for games. CRC Press.

[Lew02] Lewis, M., Jacobson, J. (2002). Game engines. Communications of the ACM, 45(1), 27.

[Doh03] Doherty, M. (2003). A software architecture for games. University of the Pacific Department of Computer Science Research and Project Journal (RAPJ), 1(1).

[Woo02] Wooldridge, M. (2002). An introduction to multiagent systems. John Wiley Sons.

[Gla06] Glavic, M. (2006). Agents and multi-agent systems: a short introduction for power engineers.

[Pos07] Poslad, S. (2007). Specifying protocols for multi-agent systems interaction. ACM Transactions on Autonomous and Adaptive Systems (TAAS), 2(4), 15.

[Sil03] Silva, C. T., Castro, J., Tedesco, P. A. (2003). Requirements for Multi-Agent Systems. WER, 2003, 198-212.

[Pon13] Pons, L., Bernon, C. (2013, October). A Multi-Agent System for Autonomous Control of Game Parameters. In Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on (pp. 583-588). IEEE.

[Fin08] Finnsson, H., Bjornsson, Y. (2008, July). Simulation-Based Approach to General Game Playing. In Aaai (Vol. 8, pp. 259-264).

[Gra13] Grant, M., Sandeep, V., Fuhua, L. (2013, November). Integrating Multiagent Systems into Virtual Worlds. In 3rd International Conference on Multimedia Technology (ICMT-13). Atlantis Press.

[Jep10] P. Jepp, M. Fradinho and J. M. Pereira, "An Agent Framework for a Modular Serious Game," 2010 Second International Conference on Games and Virtual Worlds for Serious Applications, Braga, 2010, pp. 19-26.

 ISBN 978-80-86943-38-1

[Dig09] Dignum, F., Westra, J., van Doesburg, W. A., Harbers, M. (2009). Games and agents: Designing intelligent gameplay. International Journal of Computer Games Technology, 2009.

[Bec14] Becker-Asano, C., Ruzzoli, F., Holscher, C., Nebel, B. (2014). A multi-agent system based on unity 4 for virtual perception and wayfinding. Transportation Research Procedia, 2, 452-455.

[Ado01] Adobbati, R., Marshall, A. N., Scholer, A., Tejada, S., Kaminka, G. A., Schaffer, S., Sollitto, C. (2001, May). Gamebots: A 3d virtual world test-bed for multi-agent research. In Proceedings of the second international workshop on Infrastructure for Agents, MAS, and Scalable MAS (Vol. 5, p. 6). Montreal, Canada.

[Sac11] Sacerdotianu, G., Ilie, S., Badica, C. (2011, September). Software Framework for Agent-Based Games and Simulations. In 2011 13th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (pp. 381-388). IEEE.

[Ara08] Aranda, G., Carrascosa, C., Botti, V. (2008, September). Characterizing massively multiplayer online games as multi-agent systems. In International Workshop on Hybrid Artificial Intelligence Systems (pp. 507-514). Springer, Berlin, Heidelberg.

[Ara12] Aranda, G., Trescak, T., Esteva, M., Rodriguez, I., Carrascosa, C. (2012). Massively multiplayer online games developed with agents. In Transactions on Edutainment VII (pp. 129-138). Springer, Berlin, Heidelberg.

[Gar06] Garces, A., Quiros, R., Chover, M., Camahort, E. (2006). Implementing moderately open agent-based systems. In IADIS International Conference WWW/Internet 2006 (pp. 360-369).

[Gar07] Garces, A., Quiros, R., Chover, M., Huerta, J., Camahort, E. (2007, February). A development methodology for moderately open multi-agent systems. In Proceedings of the 25th conference on IASTED International Multi-Conference: Software Engineering, SE (Vol. 7, pp. 37-42).

[Gar10] Garces, A., Quiros, R., Chover, M., Camahort, E. (2010). Implementing virtual agents: a HABA-based approach. Int J Multimed Appl, 2, 1-15.

[Rem15] Remolar, I., Garces, A., Rebollo, C., Chover, M., Quiros, R., Gumbau, J. (2015). Developing a virtual trade fair using an agent-oriented approach. Multimedia Tools and Applications, 74(13), 4561-4582.

[Fag95] Fagin, R., Halpern, J. Y., Moses, Y., Vardi, M. Y. (1995). Reasoning about knowledge MIT Press. Cambridge, MA, London, England.

[Rus95] Russell, S. J., Subramanian, D. (1994). Provably bounded-optimal agents. Journal of Artificial Intelligence Research, 2, 575-609.

[Gen87] Genesereth, M. R., Nilsson, N. J. (1987). Logical foundations of artificial. Intelligence. Morgan Kaufmann, 2.

 ISBN 978-80-86943-38-1

Real Time Pedestrian and Object Detection and Tracking-based Deep Learning: Application to Drone Visual Tracking

R. Khemmar
Institute for Embedded Systems Research, ESIGELEC*, UNIRouen, Normandy University*, Saint Etienne du Rouvray, 76800, France
redouane.khemmar@esigelec.fr

M. Gouveia
Institute for Embedded Systems Research, ESIGELEC*, UNIRouen, Normandy University*, Saint Etienne du Rouvray, 76800, France
gouveia.matthias@hotmail.com

B. Decoux
Institute for Embedded Systems Research, ESIGELEC*, UNIRouen, Normandy University*, Saint Etienne du Rouvray, 76800, France
benoit.decoux@esigelec.fr

JY. Ertaud
Institute for Embedded Systems Research, ESIGELEC, UNIRouen, Normandy University, Saint Etienne du Rouvray, 76800, France
jean-yves.ertaud@esigelec.fr

ABSTRACT

This work aims to show the new approaches in embedded vision dedicated to object detection and tracking for drone visual control. Object/Pedestrian detection has been carried out through two methods: 1. Classical image processing approach through improved Histogram Oriented Gradient (HOG) and Deformable Part Model (DPM) based detection and pattern recognition methods. In this step, we present our improved HOG/DPM approach allowing the detection of a target object in real time. The developed approach allows us not only to detect the object (pedestrian) but also to estimates the distance between the target and the drone. 2. Object/Pedestrian detection-based Deep Learning approach. The target position estimation has been carried out within image analysis. After this, the system sends instruction to the drone engine in order to correct its position and to track target. For this visual servoing, we have applied our improved HOG approach and implemented two kinds of PID controllers. The platform has been validated under different scenarios by comparing measured data to ground truth data given by the drone GPS. Several tests which were ca[1]rried out at ESIGELEC car park and Rouen city center validate the developed platform.

Keywords

Object detection, object recognition, visual tracking, tracking, pedestrian detection, deep learning, visual servoing, HOG, DPM.

1. INTRODUCTION

The works presented in this paper are a part of ADAPT[1] project (Assistive Devices for empowering disAbled People through robotic Technologies) which

[1]This work is carried out as part of the INTERREG VA FMA ADAPT project ”Assistive Devices for empowering disAbled People through robotic Technologies” http://adapt-project.com/index.php. The Interreg FCE Programme is a European Territorial Cooperation programme that aims to fund high quality cooperation projects in the Channel border region between France and England. The Programme is funded by the European Regional Development Fund (ERDF).

focuses on smart and connected wheelchair to compensate for user disabilities through driving assistance technologies. One of the objectives of the project is to develop an Advanced Driver-Assistance System (ADAS) platform for object detection, recognition, and tracking for wheelchair applications (object detection, obstacle avoidance, etc.). The work presented in this paper is related to object/pedestrian detection. In general, ADAS is used to improve safety and comfort in vehicles. ADAS is based on the combination of sensors (RADAR, LIDAR, cameras, etc.) and algorithms that ensure safety of vehicle, driver, passenger and pedestrian based on different parameters such as traffic, weather, etc. [26]. Here in this project, ADAS aims to detect pedestrian. Our contribution aims to develop a perception system based on object detection with different approaches such as HOG, DPM, and Deep Learning. This paper is organized as follows: Section 1 introduces the motivation of the paper. Section 2 presents the state of the art about object detection/tracking and visual control. Section 3 presents a comparison between the adopted object detection algorithms and some results obtained. Section 4 illustrates our improved HOG/DPM approach applied to object/pedestrian detection and tracking. In the same section, we present an innovative solution to estimate the distance separating objects to the vehicle or to the drone. The visual control system-based multi approach controller will be presented in Section 5. Finally, in Section 6, we will conclude this paper.

2. STATE OF THE ART AND RELATED WORK

State of the Art

Object detection is a key problem in computer vision with several applications like automotive, manufacturing industry, mobile robotics, assisted living, etc. Pedestrian detection is a particular case of object detection that can improve road security and is considered as an important ADAS component in the autonomous vehicle. In [13], a very in-depth state of the art for pedestrian detection is presented. Three main contributions are developed by the authors: 1. Study of the statistics of the size, position, and occlusion patterns of pedestrians in urban scenes in a large pedestrian detection image dataset, 2. A refined per frame evaluation methodology that allows to carry out informative comparisons, including measuring performance in relation to scale and occlusion, and 3. Evaluation of 16 pre-trained state of the art detectors [13]. As a main conclusion of the study, detection is disappointing at low resolutions and for partially occluded pedestrians. In [1], a real-time pedestrian detection with DPM algorithm applied to automotive application is presented. The system is based on a multiresolution pedestrian model and shows superior detection performance than classical DPM approaches in the detection of small pixel-sized pedestrians. The system was evaluated with the Caltech Pedestrian benchmark [2], which is the largest public pedestrian database. The practicality of the system is demonstrated by a series of use case experiments that uses Caltech video database. The discriminatively trained, multiresolution DPM is presented as an algorithm between generative and discriminative model [3][7]. The algorithm has different steps: building a pyramid of images at different scales, using several filters and part filters to get responses. The algorithm combines these different responses in a star-like model then uses a cost function, and trains classifiers by Support Vector Machine (SVM) classifier. This algorithm is still a widely used algorithm particularly in combination with DPM [8]. As another method of object detection, the Integral Channel Features (ICF) [1], can find a combination of multiple registered image channels, which are computed by linear and nonlinear transformations [9]. Integrating some features like HOG and do a training by AdaBoost in a cascade way can lead to pedestrian detection with good accuracy [9]. The sliding window methods (also called pyramid methods) are used in object detection with a high cost of detection time. In a recent work, proposing high-recall important regions is widely used [10]. In another way, the approaches based on Deep Learning, also called Convolutional Neural Networks (CNN), become very successful for feature extraction in the image classification task [33][34]. Rich Feature Hierarchies for Convolutional Neural Networks (RCNN) model [12], that combines CNN and selective search can be taken as an example. This algorithm has made a huge progress on object detection task like PASCAL VOC. It will be presented in the next section.

Related Work

In the literature, for similar tasks, several approaches have been used for object detection and pattern recognition, such as HOG/DPM, KLT/RMR (Kanade-Lucas-Tomasi/Robust Multiresolution Estimation of Parametric Motion Models) and Deep Learning [1]. For example, in Google Robot's Project [17], a deep learning model is applied to articulated robot arm for the picking up of objects. In Kitti Vision Benchmark Suite Project (KIT and Toyota Technological Institute) [18], an object detection and orientation estimation benchmark is carried out. The system allows not only localization of objects in 2D, but also estimation of their orientation in 3D [18]. In [19], an example of end-to-end object detection from Microsoft is described. For the task of detecting objects in images, recent methods based on convolutional neural networks (CNN, Deep-Learning) like SSD [18][20] allow detection of multiple objects

 ISBN 978-80-86943-38-1

in images with high accuracy and in real-time. Furthermore, the SSD model is monolithic and relatively simple compared to other models, making it easier to use for various applications. For the task of grasping objects, people from the AI-Research of Google have recently used Deep Learning to learn hand-eye coordination for robotic grasping [21]. The experimental evaluation of the method demonstrates that it achieves effective real-time control, and it can successfully grasp new objects, and correct mistakes by continuous servoing (control). In many applications of detection of objects like pedestrian, cyclists and cars, it is important to estimate their 3D orientation. In outdoor environments, solutions based on Deep Learning have been recently shown to outperform other monocular state-of-the-art approaches for detecting cars and estimating their 3D orientation [22][23]. In indoor environments, it has been shown that using synthetic 3D models of objects to be detected in the learning process of a CNN can simplify it [23]. In [24] and [25], we can find an evaluation of the state of the art object (pedestrian) detection approaches based on HOG/DPM. In [31], B. Louvat *et al*. have presented a double (cascade) controller for drone-embedded camera. The system is based on two aspects: target position estimation-based KLT/RMR approaches and control law for the target tracking. The developed platform was validated on real scenarios like house tracking. In [32], B. Hérissé has developed in his PhD thesis an autonomous navigation system for a drone in an unknown environment based on optical flow algorithms. Optical flow provides information on velocity of the vehicle and proximity of obstacles. Two contributions were presented: automatic landing on a static or mobile platforms and field following with obstacle avoidance. All algorithms have been tested on a quadrotor UAV built at CEA LIST laboratory.

3. OBJECT DETECTION APPROACH

In order to identify the most suitable approach to our object and/or pedestrian detection application for the autonomous vehicle (a drone in this study), we need to establish the feasibility of several scientific concepts in pattern recognition approaches such as the KLT/RMR, SIFT/SURF, HOG/DPM but especially recent approaches of the artificial intelligence field, such as Deep-Learning. In this paper, we will focus on object detection and tracking-based through improved HOG/DPM approaches.

Classical Approaches

We have started out our experimentations by implementing point of interest approaches like Scale Invariant Feature Transform (SIFT) [9] and Speeded-Up Robust Features (SURF) [35], based on descriptors that are very powerful to find matching between images and to detect objects. These methods allow the extraction of visual features which are invariant to scale, rotation and illumination. Despite their robustness to changing perspectives and lighting conditions, we found that the approach is not robust for object/pedestrian detection. SURF is better as it is up to twice as fast as SIFT. However, SIFT is better when a scale change or an increase in lighting is applied to the image. For pedestrians detection, SIFT and SURF therefore remain insufficient for our application. We have also experimented KLT approach [15] for extraction of points of interest and tracking them between an image taken at t-1 and another image taken at t. The approach has high accuracy and is fast, but on the other hand, it is not robust to perturbations, for example when the target is displaced too much, and in case of highly textured images. This is why we have decided to experiment the RMR approach [16], which has very low precision but is considered to be very robust. We have obtained results which are same as KLT approach. A hybrid KLT/RMR approach would be a better solution where we can have benefits of both the approaches.

Object Detection-based HOG/DPM

HOG and DPM algorithms first calculate image features. They apply classifiers on databases of positive images (with pedestrian) and negative images (without pedestrian). They have the advantage of being accurate and give relatively good results; however, their calculating time is high. We have therefore experimented three approaches: HAAR, HOG and DPM.

3.1.1 Pseudo-HAAR Features

HAAR classifiers use features called pseudo-HAAR [36][27]. Instead of using pixel intensity values, the pseudo-HAAR features use the contrast variation between rectangular and adjacent pixel clusters. These variations are used to determine light and dark area. To construct a pseudo-HAAR feature, two or three adjacent clusters with relative contrast variation are required. It is possible to change the size of features by increasing or decreasing the pixel clusters. Thus, it makes it possible to use these features on objects with different sizes.

3.1.2 HOG Algorithm

HOG is a descriptor containing key features of an image. These features are represented by the distribution of image gradient directions. HOG is an algorithm frequently used in the field of pattern recognition. It consists of five steps:

1. Sizing of the calculation window ; by default the size of the image to be processed is 64x128 pixels
2. Calculation of image gradients using simple masks

ISBN 978-80-86943-38-1

3. Image division of 64x128 into 8x8 cells. For each cell, HOG algorithm calculates the histogram.
4. Normalization of histograms by 16x16 blocks (*ie.* 4 cells)
5. Calculation of the size of the final descriptor.

The obtained descriptor is then given to a SVM classifier. The classifier needs many positive and negative images. The mixture HOG/SVM gives good results with limited computing resources.

3.1.3 DPM Algorithm

The hardest part for object detection is that there are many variances. These variances arise from illumination, change in viewpoint, non-rigid deformation, occlusion, and intra-class variability [4]. The DPM method is aimed at capturing those variances. It assumes that an object is constructed by its parts. Thus, the detector will first find a match by coarser (at half the maximum resolution) root filter, and then using its part models to fine-tune the result. DPM uses HOG features on pyramid levels before filtering, and linear SVM as a classifier with training to find the different part locations of the object in the image. Recently, new algorithms have been developed in order to make DPM faster and more efficient [4][8].

Pedestrian Detection-based HAAR/HOG/DPM

We have carried out all the experiments with 6 different databases dedicated to the pedestrian detection: ETH, INRIA, TUD Brussels, Caltech, Daimler and our own ESIGELEC dataset dedicated to the pedestrian detection. Fig. 1 and Fig.2 shows respectively results obtained under ETH and ESIGELEC datasets.

Figure 1. Comparison between HOG and DPM applied under ETH dataset (640x480-image resolution). Left Column: HOG algorithm, Right Column: DPM algorithm

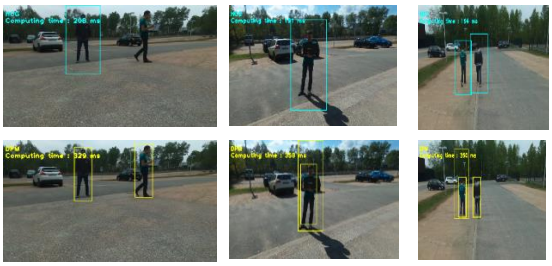


Figure 2. Comparison between HOG and DPM applied under ESIGELEC dataset (640x480-image resolution). Top row: HOG algorithm, Bottom row: DPM algorithm.

Object Detection-based Deep Learning

Many detection systems repurpose classifiers or localizers to perform detection. They apply the classification to an image at multiple locations and scales. High scoring regions of the image are considered as positives detections [4][6]. CNN classifier, like RCNN, can be used for this application. This approach gives good results but require many number of iterations to process a single image [4][6]. Many detection algorithms using selective search [4][5] with region proposals have been proposed to avoid exhaustive sliding window. With deep learning, the detection problem can be tackled in new ways, with algorithms like YOLO and SSD. We have implemented those two algorithms for object detection in real time [14], and obtained very good results (qualitative evaluation). Fig. 3 shows an example of our deep learning model applied in pedestrian detection within ESIGELEC dataset at the city center of Rouen.

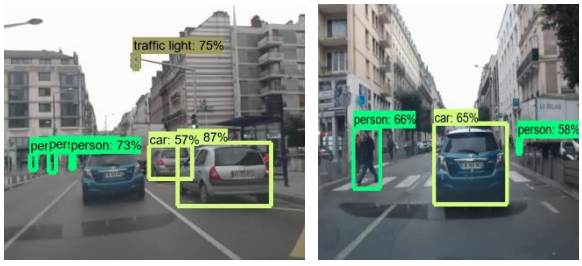


Figure 3. Object/Pedestrian detection-based deep learning (Yolo and SSD). The results have been obtained under GPU Nvidia Quadro K4000M machine.

ISBN 978-80-86943-38-1

4. OBJECT DETECTION AND TRACKING-BASED FASTER DPM ALGORITHM

Improved HOG Algorithm

We have identified four methods to improve HOG/DPM algorithm, which correspond to the five step of the HOG algorithm. In order to be able to make comparisons with the original version, a classifier was trained on the INRIA and TUD-Brussels image datasets. The number of available images in these datasets is relatively small compared to some other datasets, which has an impact on the pedestrian detection quality. The objective is to improve the quality of HOG/DPM pedestrian detection and to significantly reduce the computation time for the detection in road scenes. The tests were carried out on the ESIGELEC dataset; 1. Gamma Correction: in order to improve the pedestrian image quality, we performed a gamma pre-processing on the images to be trained and tested. Gamma correction can illuminate more the dark areas in images if gamma coefficient is greater than 1 or, in the contrary, darken them if the coefficient is less than 1. We found fewer parasites in the processed images. 2. Image Resizing: HOG algorithm performs calculations on images of size 64x128 pixels. We performed calculation with windows size of 128x256. By doubling the size of the calculation window, we have improved the accuracy but also doubled the computation time (for example from 58ms in the classic HOG to 115ms for the improved HOG). 3. Negative Gradients: when calculating gradients, improved HOG uses negative gradients (180° to 0) and positive gradient (0 to 180°) like classical HOG. This allows the calculation of histogram with 18 values (9 values in classic HOG). The calculation time does not vary, however, by taking the negative gradients (signed gradient) into account, a small improvement was carried out but considered as not significant. However, we found presence of noise, which does not improve the classic HOG. 4. Normalization Vector: as a last step, HOG performs a standardization of 2x2 cells, *ie*. 32x32 pixels with a pitch of 16x16. The object detection is degraded and the computation time doubles, which cannot considered as an improvement of classic HOG.

Faster-DPM Algorithm

The DPM algorithm is applied on the entire image and this is done at each iteration of the pedestrian video sequence. In order to reduce the computing time, the idea is to apply HOG algorithm only in a Region of Interest (RoI) in which the target (here the pedestrian) is located. This will drastically reduce the computing time and will also better isolate the object. Firstly, the DPM is applied all over the image once to locate the object. Secondly, and after obtaining a RoI surrounding the object to be detected as a bounding box (yellow box in figure 4), we built a New RoI (NRoI) by providing a tolerance zone (a new rectangle like the green one in figure 4 which is larger than the first one). Starting form the second iteration, DPM algorithm is applied only in this new image, which is represented by NRoI. If the target to be detected is lost, the DPM algorithm is re-applied over the entire image. In Fig. 4, we can see that Faster-DPM improves target detection by reducing time from 4 to 8 times less than classic DPM.

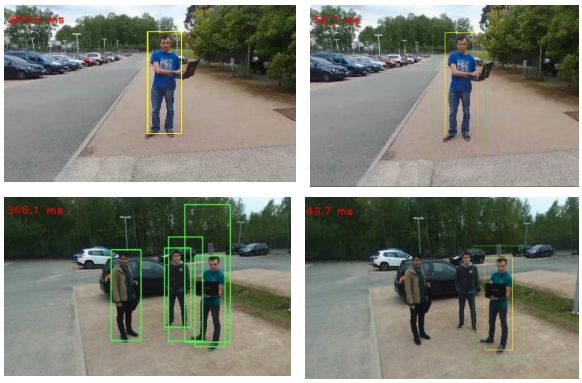
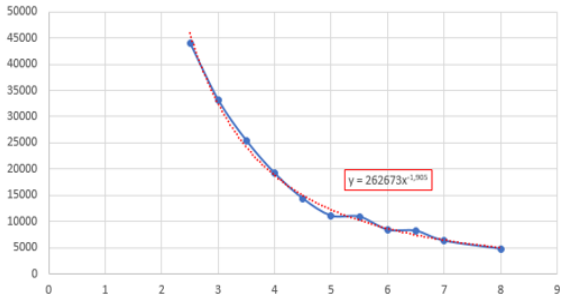


Figure 4. Comparison between classic DPM and Faster-DPM. Left top and bottom image: object detection-based DPM, Right top and bottom image: object detection-based Faster-DPM with adaptive RoI (green box).

Pedestrian Distance Estimation

To have a better information on the detected pedestrians, it is necessary to estimate the distance separating the pedestrians from the vehicle. As our system is based on a monocular camera, the measurement of this distance has to be estimated. The law called “Inverse Squares”, used in astronomy to calculate the distance between stars, inspired us: “*physical quantity (energy, force, etc.) is inversely proportional to the square of the distance of stars”*. By analogy, the physical quantity represents the area of our RoI (bounding box surrounding the target). We have used a parametric approach. We have taken measurements at intervals of 50 cm, and the corresponding object RoI surfaces have been recorded. Fig. 5 shows the result obtained.



ISBN 978-80-86943-38-1

Figure 5. Detected RoI surface (blue curve) *vs* distance from the object (red curve): y axis represents the surface (pixels^2), and x axis represents distance (m).

The blue curve looks like a square root. This is validated by the trend curve, whose as equation (1) is:

$$y = 262673 * x^{-1.905} \quad (1)$$

where x is the abscissa and y the ordinate.
The equation of blue curve is (2):

$$S = A * d^{-2} \quad (2)$$

where S is the surface in pixel^2, d is the distance to be estimated, and A = 262673.

The ideal detection zone is between 3 and 7 meters with an error of 5%. The Tab. 1 illustrates the calibration process carried out for the measurement environment.

Distance (m)	RoI Surface (pixel^2)	K = S * d^{-2}
2.5	44044	275275
3	33180	298620
3.5	25392	311052
4	19200	307200
4.5	14360	290790
5	11102	277550
5.5	10920	330330
6	8427	303372
6.5	8216	347126
7	8348	311052
8	4800	307200

Table 1. Measurement environment calibration.

5. VISUAL CONTROL-BASED MULTI APPROACH CONTROLLER

Drone Double Controller

The speed servo control of the drone allows it to track the target continuously and in real time. The image processing as a closed loop gives the coordinates of the target in (x,y,z) plan. Using this information, the system send instruction to the Drone engines corresponding to the real time position of the target in order to correct the position of the drone. We have developed two kinds of controllers: Proportional-Integral-Derivative (PID) and Adaptive Super Twisting (AST) controller [28]. The controller system is based on three corrections: 1. Drone altitude correction, 2. Drone rotation speed, and 3. Drone navigation. Under Robot Operating System (ROS) system, the instructions are considered as speeds sent to drone engines.

Firstly, we have applied a Proportional (P) controller:

- Altitude : $K_p = 0.004$
- Forward translation : $K_p = 0.08$

Secondly, we have applied a classic PID controller (3):

$$V_{rotation} = K_p * erreur + K_i * \int_0^t erreur * dt + K_d * \frac{d}{dt}(erreur) \quad (3)$$

With Rotation: $K_p = 0.0041$, $K_i = 0.0003$, $K_d = 0.0003$

Lastly, in (4), we have applied an AST (nonlinear PID) controller:

$$V_{rotation} = K_p * \left(\frac{erreur}{|erreur| + A}\right) * \sqrt{|erreur|} + K_i * \int_0^t \frac{erreur}{|erreur| + A}\, dt \quad (4)$$

with $A = 0.001$, $K_p = 0.013$, and K_i=0.02.

The P controller is amply enough to enslave the altitude of the drone and its translation. However it is the rotational slaving that predominates; we need to keep the target in the center of the image and the target moves strongly from right to left and vice versa.

Overall, the visual servoing that we have adopted uses the data extracted during the HOG/DPM or deep learning image processing (visual controller) and sends commands to the drone actuators. This is a double controller: speed control (internal loop) to control the motors and servo positioning (external loop) to position the drone according to the target position to track. The correction can be done with classical correctors (P, PI, PID) or more advanced commands like for example AST controller. Fig. 6 illustrates the architecture of our cascade control used.

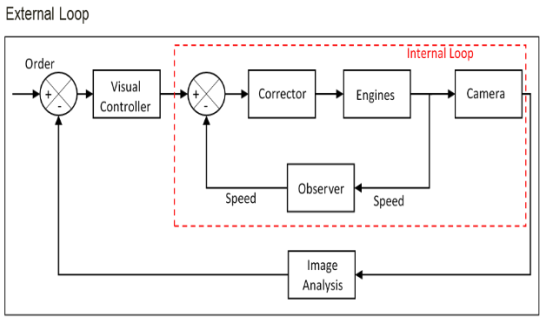


Figure 6. Double controller with external (black) and internal loop (in red).

ISBN 978-80-86943-38-1

Test & Validation

Indoor and outdoor environment tests were carried out throughout the project. For the final test phases, scenarios have been established. In order to validate the final platform, several scenarios has been defined. The platform gives very good results. Better results are obtained when the target (person) is alone in the environment. Performance is reduced when the target is in a group, despite the use of an adaptive region of interest. In addition, there is always the risk of losing the target, object detection algorithms do not have a detection rate of 100% and sometimes the person to track is no longer detected. If the target is no longer detected, the drone is ordered to switch to stationary mode as a filled situation. We are correcting this problem through two different methods: 1. An estimation of the future position of the target based on improved Kalman filter that gives better results; from now on, we are able to predict the position of the target to be followed and thus to minimize a possible confusion of this target with another target of the same nature (as for example the case of several people who walk together). A second approach is also under development which concerns deep learning not only for object/people detection and tracking, but also for object distance and orientation estimation. To compare the performance of the two implemented control laws (PID controller and AST controller), the GPS coordinates of each trajectory were recorded and compared. In order to illustrate the results obtained, the target (here a person) has made a reference trajectory in the form of a square. Fig. 7 shows the performances obtained on a square shape trajectory. We can see that the PID controller is more accurate than the AST controller.

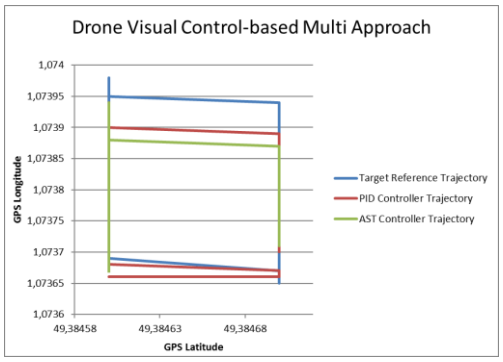
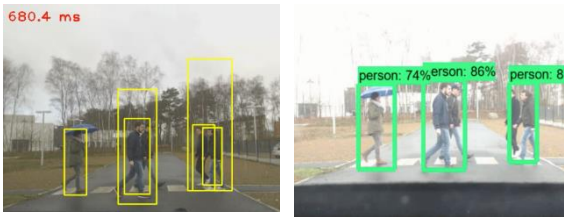


Figure 7. PID and AST comparison in Object/Person detection and tracking-based on our improved HOG/DPM. Top image: target square trajectory (blue line) carried out but the mobile target (person). Bottom image: comparison for the different trajectory carried out by person (ground truth data), PID controller and AST controller. x and y axis represents the geographic coordinate system coordinates.

6. EXPERIMENTAL RESULTS

The developments were carried out on an Ubuntu Linux platform with ROS, which is a set of computer tools for the robotics environment that includes a collection of tools, libraries and conventions that aim to simplify the process of operation, thus allowing more complex and robust robot behavior [29]. The ROS architecture developed comprises 4 different nodes: Node 1: image acquisition from the drone's embedded camera, Node 2: image processing and calculation of the position of the target to follow, Node 3: visual servoing-based on PID or AST controller, and Node 4: Sending commands to the drone (manual and/or autonomous steering). The drone used is a Parrot Bebop 2 with 22 minutes of autonomy, weight 420g, range of 300m, camera resolution of 14 Mpx, and maximum speed of 13m/s. We have carried out several tests with six datasets dedicated to pedestrian detection. In this section, we present the tests carried out on the ESIGELEC dataset including tests scenarios under real traffic conditions in Rouen shopping center car park and Rouen city center. The results obtained are shown in Fig. 8.



ISBN 978-80-86943-38-1

Figure 8. Pedestrian Detection with improved HOG/DPM approach under ESIGELEC Indoor/Outdoor dataset. Left top image: Pedestrian detection with calculated distance under DPM, Right top image: Pedestrian detection-based SSD deep learning model, Left bottom image: Pedestrian detection-based DPM, Right bottom image: Pedestrian detection-based Faster-DPM.

7. CONCLUSION

In this paper, we have presented a new approach for object detection and tracking applied for drone visual control. A comparison study between different approaches dedicated to pattern recognition and object/pedestrian detection has been carried out. We have present our contribution to improve the quality of both HOG and DPM algorithms. We have also implemented deep learning based Yolo and SSD algorithm for pedestrian detection. An innovative method for distance calculation of pedestrian/object is presented in this paper. The approach were validated within experiments and the accuracy is 5% of errors. The system detects not only object/target to follow but also their distance from the drone. The system is generic so that it is "*applicable*" on any type of platform and/or environment (pedestrian detection for autonomous vehicle and smart mobility, object detection for smart wheelchair, object detection for autonomous train, etc.). This work aims to show the feasibility of scientific and technological concepts that affect object detection and tracking-based classic approaches like HOG/DPM, but also deep learning approaches-based Yolo or SSD. We have validated the whole development under several scenarios by using both parrot drone platform and real vehicle in real traffic conditions (city center of Rouen in France).

8. ACKNOWLEDGMENTS

This research is supported by ADAPT project (This work is carried out as part of the INTERREG VA FMA ADAPT project ”Assistive Devices for empowering disAbled People through robotic Technologies” http://adapt-project.com/index.php. The Interreg FCE Programme is a European Territorial Cooperation programme that aims to fund high quality cooperation projects in the Channel border region between France and England. The Programme is funded by the European Regional Development Fund (ERDF). Many thanks to the engineers of Autonomous Navigation Laboratory of IRSEEM for the support during platform tests.

9. REFERENCES

[1]. H. Cho, P. E. Rybski, A. B-Hillel, W. Zheng.: Real-time Pedestrian Detection with Deformable Part Models, 2012

[2]. Caltech Pedestrian Detection Benchmark Homepage, http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/, last accessed 2018/01/14.

[3]. Felzenszwalb, P., McAllester, D., & Ramanan, D. A discriminatively trained, multiscale, deformable part model. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on (pp. 1-8). IEEE (June 2008).

[4]. Jong-Chyi Su.: State of the Art object Detection Algorithms. University of California, San Diego, 9500 Gilman Dr. La Jolla, CA., 2014.

[5]. Koen E. A. van de Sande, Jasper R. R. Uijlings, TheoGevers, Arnold W. M. Smeulders, Segmentation As Selective Search for Object Recognition, ICCV, 2011.

[6]. Alex Krizhevsky , Ilya Sutskever , Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks, NIPS, 2012.

[7]. Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, CVPR, 2014.

[8]. P. Dollar, C. Wojek, B. Schiele.: Pedestrian Detection: An Evaluation of the State of the Art. IEEE Transactions on Pattern Analysis and Machine Intelligence,Volume: 34, Issue: 4, April 2012.

[9]. D. G. Lowe. Distinct Image Features from Scale-Invariant Keypoints. Computer Science Department. University of British Columbia. Vancouver, B. C. Canada. January, 5, 2004 28.

[10]. Kanade-Lucas-Tomasi. KLT Feature Tracker. Computer Vision Lab. Jae Kyu Suhr. Computer Vision (EEE6503) Fall 2009, Yonsei Uni.

[11]. J. M. Odobez and P. Bouthemy. Robust Multiresolution Estimation of Parametric Motion Models. IRIS/INRIA Rennes, Campus de Beaulieu, February, 13, 1995.

[12]. Object Detection Homepage, http://cseweb.ucsd.edu/~jcsu/reports/ObjectDetection.pdf, last accessed 2018/01/14.

[13]. Yan, J., Lei, Z., Wen, L., & Li, S. Z. The fastest deformable part model for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2497-2504). 2014.

 ISBN 978-80-86943-38-1

[14]. Yolo Homepage, https://pjreddie.com/darknet/yolo/, last accessed 2018/01/19.

[15]. Koen E. A. van de Sande, Jasper R. R. Uijlings, Theo Gevers, Arnold W. M. Smeulders, Segmentation As Se-lective Search for Object Recognition, ICCV, 2011.

[16]. Alex Krizhevsky , Ilya Sutskever , Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. NIPS, 2012.

[17]. Dave Gershgon, Google's Robot Are Learning How to Pick Things Up. Popular Science, March 8, 2016.

[18]. Andreas Geiger, Philip Lenz, Christoph Stiller, Raquel Urtasun. Object Detection Evaluation. The Kitti Vision Benchmark Suite. Karlsruhe Institute of Technology. IEEE CVPR, 2012.

[19]. Eddie Forson. Understanding SSD MultiBox-Real Time Object Detection in Deep Learning. Towards Data Science. November, 18th. 2017.

[20]. Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C.-Y., Berg A. C., (2016, déc.). Single-Shot Multibox Detector. https://arxiv.org/abs/1512.02325.

[21]. Levine S., Pastor P., Krizhevsky A., Ibarz J., Quillen D. (2017, June). Learning Hand-Eye Coordination for Robotic Grasping with DeepLearning and Large-Scale Data Collection. The International Journal of Robotics Research.

[22]. Chabot F., Chaouch M., Rabarisoa J., Teulière C., Château T.,. (2017, July). Deep MANTA: A Coarse-to-fine Many-Task Network for joint 2D and 3D vehicle analysis from monocular image. IEEE Conference on Computer Vision and Pattern Recognition.

[23]. Mousavian A., Anguelov D., Flynn J., Kosecka J. (2017, July). 3D Bounding Box Estimation Using Deep Learning and Geometry. IEEE Conference on Computer Vision and Pattern Recognition.

[24]. Georgakis G., Mousavian A., Berg A. C., Kosecka J. (2017, July). Synthesizing Training Data for Object Detection in Indoor Scenes. IEEE Conference on Computer Vision and Pattern Recognition

[25]. P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian Detection: A Benchmark. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Piscataway, NJ, pp. 304-311. ISBN 978-1-4244-3991-1. 2009.

[26]. Junjie Yan, Xucong Zhang, Zhen Lei, Shengcai Liao, Stan Z. Li. Robust Multi-Resolution Pedestrian Detection in Traffic Scenes. Computer Vision Foundation. CVPR 2013.

[27]. http://www.bmw.com/com/en/insights/technology/efficientdynamics/phase_2/, last accessed 2018/01/14.

[28]. P. Viola, M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. Computer Vision and Pattern Recongition Conferencies. 2001.

[29]. Mohamed, G., Ali, S. A., & Langlois, N. (2017). Adaptive Super Twisting control design for manufactured diesel engine air path. The International Journal of Advanced Manufacturing Technology, 1-12. 2017.

[30]. ROS homepage: http://www.generationrobots.com/blog/fr/2016/03/ros-robot-operating-system-3/. Last accessed 2017/12/12.

[31]. B. Louvat, "Analyse d'image pour l'asservissement d'une camera embarquée dans un drone". Gipsa-lab, February, 5th, 2008, Grenoble. 2008.

[32]. B. Hérissé, "Asservissement et navigation autonome d'un drone en environnement incertain par flot optique. PhD thesis, Université Sophia Antipolis, UNSA et I2S CNRS, November 19th, 2010.

[33]. R. Girshick, "Fast R-CNN," ICCV, 2015

[34]. S. Ren et al., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," arXiv:1506.01497. 2015.

[35]. H. Bay, T. Tuytelaars, and L. V. Gool. "SURF: Speeded Up Robust Features", Computer Vision – ECCV 2006, pp. 404-417. 2006.

[36]. D. Gerónimo, A. López, D. Ponsa, A.D. Sappa. "Haar wavelets and edge orientation histograms for on–board pedestrian detection". In: Pattern Recognition and Image Analysis, pp. 418-425. 2007.

 ISBN 978-80-86943-38-1

ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

Title Segmentation in Arabic Document Pages

Hassina Bouressace
University of Szeged,
13 Dugonics square,
Szeged, H-6720 Hungary
bouressacehassina@hotmail.fr

ABSTRACT

Recent studies on text line segmentation have not focused on title segmentation in complex structure documents, which may represent the upper rows in each article of a document page. Many methods cannot correctly distinguish between the titles and the text, especially when it contains more than one title. In this paper, we discuss this problem and then present a straightforward and robust title segmentation approach. The proposed method was tested on PATD (Printed Arabic Text Database) images and we achieved good results.

Keywords

Arabic language, Projection Profile, Text-line segmentation, Connected Components

1 INTRODUCTION

The goal of Document Analysis and Recognition (DAR) [Sim08] is the automatic detection and extraction the information existing on a page, where the output of DAR systems is usually presented in a structured symbolic that can then be processed by a computer. The principle of DAR is closely related to official documents such as (newspapers, business letters, books, and journals), where the information of these documents is presented in digital form such as a PDF, HTML or via a digital camera containing textual information.

The type of document structure may be a simple structure or complex one, where the identification of the structure is based on the amount of information contained in the document and the way this information is presented. A document with a complex structure is usually composed of textual heterogeneous blocks, which may contain mathematical expressions, tables, graphs and, pictures [Azo95]. They may be characterized by variability of positioning, shape, and appearance of areas, in which the different text blocks are not perfectly aligned, with a complex layout that can have multiple columns with different sizes, an irregular body, and spacing.

The two families of composite documents are complex with structurally stable documents (form, commercial letter, etc.) and complex with variable structure documents in which text is found between blocks and in others (newspapers, documents, magazines, flyers, etc.). They have rich typography and are not composed solely of text but a combination, in a variable arrangement, of texts, graphics, and images. Figure 1 shows two examples of images of documents with a complex structure. With the availability of the high-resolution scanning devices along with robust computers, the Optical Character Recognition (OCR) systems can handle numerous recognition tasks of text-images. Basic existing research used text-lines as input entities for an OCR system [Gra09], hence text-line extraction plays a key role in the OCR process and its facilitation. This has led to a lot of research on improving text-line segmentation over the years [Ray15, Ryu14].

(a) (b)

Figure 1: Examples of documents with a complex structure (composite): (a)A complex structure in stable document form, (b)A complex structure in variable document form (newspaper page).[Mon11]

The application of text-line segmentation is not always easy to do, due to the existence of skew, script variations, noise, text-lines with different sizes and different fonts. One especially problematic issue, which is the aim of the study, is the line segmentation of a large scale heading, in such a way that we can present it as titles and their subtitle detection in Arabic document pages. The existing approaches for text-line extraction cannot correctly distinguish the titles from the text, especially when it contains more than one

title. The real text in documents often contains titles and subtitles, and such text lines cannot be precisely identified with state-of-the-art methods.

The state-of-the-art approach described in [Mun17] could not extract Arabic text documents with large-scale headings and titles; moreover, it is not efficient in the case of a document with a complex structure. This inspired us to develop a new method that can extract not one title, but every title and subtitle on a document page. In this paper, we present a new text-line detection method for complex-structured documents where the detected text is treated as a title or subtitle and each page contains many titles corresponding to the article numbers. The paper is organized as follows: In Section 2, the related work is described. In Section 3, we describe each step of the algorithm in detail. Experiments and results are presented in Section 4 and finally, in Section 5 we outline our conclusions and plans for the future.

2 RELATED WORK

A wide variety of title detection methods for documents can be classified and incorporated in many techniques: Active Contour Model (Snake), Horizontal Projection Profile (HPP), Vertical Projection Profile (VPP), Connected Components (CCs), the Bounding box-based method, smearing method, the Hough Transform (HT), or use of HMMs. Here, the main studies of text-line detection methods are outlined.

Bukhari et al. [Buk08] presented a robust text-line segmentation approach against skew, curl and noise, which is based an active contour model (Snake) with the novel idea of several baby snakes and their convergence in a vertical direction using the ridges which are found by applying multi-oriented anisotropic Gaussian filter banks, hence it is computationally expensive. In [Zek11, Che01], they applied horizontal projection profile (HPP) and vertical projection profile (VPP) techniques for the text-line segmentation approach by finding the inter-line gap and taking into account the separation between two consecutive lines. In [Bou18], they applied a horizontal projection, commencing with a calculation of the histogram of each block to extract the local minima by using a threshold value, and conflict resolution for assigning the existing black pixels in the separator zones to the nearest line of text by a proximity analysis. However, this method is limited to specific structures. In [Sou10], they used the Bounding box-based method where a histogram is created, then they specified the lines that have the minimum number of pixels. Afterward, the boundaries of each line were detected by determining the centroid by measuring the regional properties. In [Hus15, Alj12, Bro13], they applied a smearing method; namely smearing the consecutive black pixels in the horizontal projection, then the pixels between them were marked in black if the distance between any two was less than a threshold value. However, it fails when there is no space between two consecutive lines or overlapping lines.

3 DESCRIPTION OF THE METHOD

Titles are the key elements of documents because there are no page documents without titles and subtitles. The size of these titles is not always larger than other text on the page, especially when the title belongs to a small article. Nevertheless, subtitles are usually found above or below the title where the space and the size between the subtitle and the article text are identical. These generic characteristics present challenges in the Arabic language in terms of text-line extraction from a document page. Figure 2 illustrates the problem where the spaces between the peaks did not give us useful information for title extraction.


في تقرير رفعه إتحاد الفلاحين إلى رئيس الجمهورية

الولاة، رؤساء الدوائر والأميار وراء نهب الأراضي الفلاحية!

■ ولاة العاصمة، عنابة وهران وبومرداس على رأس قائمة المتهمين

رفعت الأمانة العام للإتحاد الوطني للفلاحين، تقريرا أسود إلى الرئيس، عبد العزيز بوتفليقة، من خلال رئيس مجلس الأمة، تشكو فيه التجاوزات والتلاعب الحاصل من طرف السلطات المحلية عبر كافة ولايات الوطن وتناشده من أجل التدخل العاجل لوضع حد لمثل هذه التجاوزات.

حبيبة محمودي

3.1 Pre-processing

We used global thresholding to produce a clear image that simplifies the processing of the later stages. In this stage, the Otsus binarization method [Zek11] is used to transform the image into two possible pixel values (0 and 1) to reduce the noise and overcome the illumination issue that arises during the scanning process.

3.2 Removing Figures and black blocks

We know of course that document pages may contain one or more figures. These figures consist of the largest proportion of pixels in some cases that give us imprecise information and this could lead to poor results in the subsequent steps. Moreover, the existence of black blocks could corrupt the essential parts that are needed later on. To overcome these problems and facilitate title and subtitle segmentation, we used formulas applied constraints on the size of the connected components, the ratio of height and width, and the density of black pixels in the connected component. In Figure 3, we give an example of figure deletion and black block reverse.

ISBN 978-80-86943-38-1

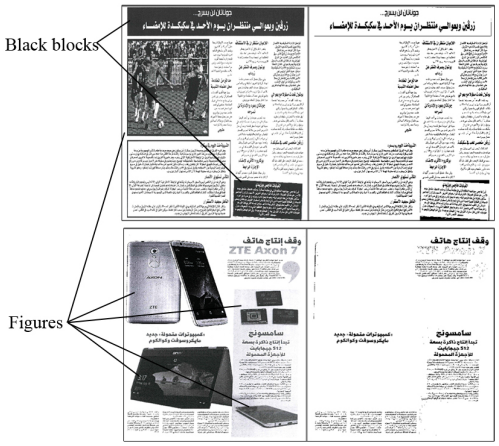


Figure 3: Examples of removing figures and black blocks.

3.3 Title Segmentation

The detection of the titles is done by taking into account the fact that not just the height of the titles is important but also the number of pixels in each component and its position. Here, our proposed method is based on RLSA and the Connected Components (CCs) technique. HRLSA, or the (Horizontal Run length smoothing algorithm)[Gor97] is then applied to the resulting image of the preceding step to eliminate spaces between words of the same line of text and Vertical RLSA smoothing is used to connect the diacritic marks to the corresponding words. Let L_0 be a horizontal segment of unit length. The Run-Length Smoothing closing algorithm fuses nearby pixels of the binary image X by γ L_0, where γ is a size parameter.

$$RLSA(X) = X \oplus \gamma L_0 \ominus \gamma L_0 \qquad (1)$$

The horizontal and vertical smoothing thresholds were determined empirically, namely (with threshold 1=1%) and (with threshold 2=0.85%) proportional to the size of the page, respectively. Actually, the characters of the titles are usually larger than those of the lines of simple text, in this case, the threshold of the horizontal RLSA smoothing was previously not enough to connect the words of a big title. To remedy this problem, we applied a second horizontal RLSA smoothing with a larger threshold (with threshold 3= 1.55 % proportional to the size of the page) only on the parts of the image containing probable major titles. These are composed of connected components whose height is greater than (1.5 x the most common text height in the document). We then applied another labeling of the related components on the RLSA smoothed image. As the words of a single line of text (simple or title) become connected, each line of text is treated as a separate component. A related component is treated as a title if its height is greater than (1.2 x the most common text height in the document), otherwise, it is treated as a simple line of text.

Figure 4: The title segmentation results of our proposed method on Arabic text documents.

3.4 Subtitle Extraction

However, these techniques only provided us with the main titles, no subtitles. Other criteria must be used to add the other titles. For this, we combined two criteria, namely the size of the related component of the previous step and its position relative to the main titles. Here, the other titles are extracted using the projection profiles (PP) method.
Let L_1 and L_2 denote the lines of text that are above and below a main title T respectively, V_1 denote the image width and V_2, V_3 denote the height. And let:

- $V_1 = 12.5\%, V_2 = 3.35\%, V_3 = 0.07\%$.
- (x_1, y_1): the coordinates of the top left-hand corner of L_1.
- (x_2, y_2): the coordinates of the bottom right-hand corner of L_1.
- (z_1, k_1): the coordinates of the top left-hand corner of T.
- (z_2, k_2): the coordinates of the bottom right-hand corner of T.
- (x_3, y_3): the coordinates of the top left-hand corner of L_2.
- (x_4, y_4): the coordinates of the bottom right-hand corner of L_2.

ISBN 978-80-86943-38-1

The lines of text L_1 and L_2 are treated as subtitles if they satisfy the following conditions:

- height of L_1 and $L_2 <$ (Threshold 4) 1.15 % proportional to the size of the page document;
- $(z_1 - x_2 < V_1) \wedge ((|\, y_1 - k_1 \,| < V_2) \vee (|\, y_2 - k_2 \,| < V_2))$
- $((k_1 - y_1 > V_3 \vee y_2 - k_2 > V_3) \wedge (y_1 - k_1 > V_3 \vee k_2 - y_2 > V_3))$
- $(x_3 - z_2 < V_1) \wedge ((|\, k_1 - y_3 \,| < V_2) \vee (|\, k_2 - y_4 \,| < V_2))$
- $((y_3 - k_1 > V_3 \vee z_2 - y_4 > V_3) \wedge (k_1 - y_3 > V_3 \vee y_4 - k_2 > V_3))$

Figure 5: The subtitle segmentation results for an Arabic document page.

We proceed in the same way with other subtitles if they exist by letting $(x1, y1)$ be the coordinates of the upper left-hand corner of T, (x_2, y_2) be the coordinates of the lower right-hand corner of T, (z_1, k_1) be the coordinates of the upper left-hand corner of subtitle L, (z_2, k_2) be the coordinates of the bottom right-hand corner of subtitle L in a recursive way until no line satisfies these conditions.

The conditions applied for subtitle and title detection are determined by the following geometrical features:

- Height: CC bounding box height,
- Width: CC bounding box width,
- Aspect Ratio: Width divided by height,
- Solidity: Area of the CC (in pixels) divided by the area of its convex hull,
- Area: Number of pixels in the CC,
- Position: CC coordinates.

Figure 6 shows the number of document pages in each threshold where the threshold is computed by getting image information under valid conditions. Every threshold in this chart is used for title or subtitle detection, which has four tests (with proportional values of 0.85, 1, 1.15 and 1.55, where these values are found by calculating the median of proportional values of the images (the ratio is extracted using information about the dimension and size of the document image, all the titles and subtitles being on each individual page). Here, the y parameter denotes the number of images that matched this proportional value. We took the best and the highest number column for each threshold. Our results were tested on over three hundred pages to check accuracy and performance.

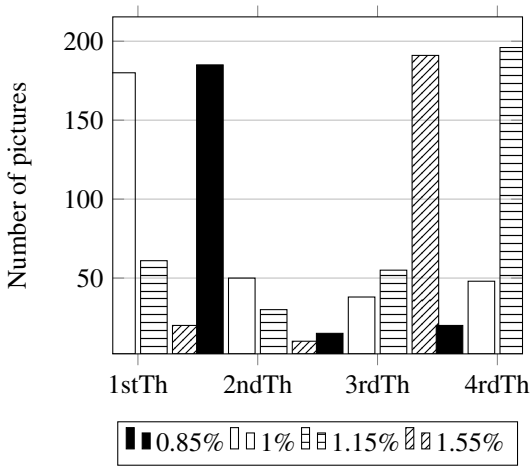


Figure 6: The number of experiments of our data thresholds.

4 EVALUATIONS AND RESULTS

Our method can be used in two modes; namely, the application returns just the cropped titles and subtitles, or it returns the whole page with colored titles and subtitles. Both demonstrate the segmentation phase in a clear way.

To evaluate our proposed system, we used the same criterion as in that described in [Ari07], which is the title-segmentation accuracy in percentage terms. The constraints are given below.

1) If a single connected component of a line is segmented to another line, this error if counted as two line errors.

2) If n subtitles and simple text are merged together, then it is counted as n line errors.

3) If n titles and subtitles are merged together, then it is counted as n line errors.

We used the following formula for computing all the errors:

$$Accuracy\% = 100 - (E / TotalTitleLines) * 100 \quad (2)$$

ISBN 978-80-86943-38-1

The algorithm was tested on three hundred scanned pages at 300 dpi got from the PATD (Printed Arabic Text Database) [Bou19], which has various styles including regular and bold, multiple font sizes and types(AL-Quds, AxTManal, Beirut, AL-Quds Bold, Kacstone, Alshrek Titles). Pages with different structures may contain one article, two articles, three articles or more. Every page has a unique format because the PATD database was collected from many documents. The algorithm gives excellent scores, which may be as high as 98.02% for titles, and 98.15% for subtitles. Table 1 below lists the results obtained during the testing process with various font types, styles, and sizes.

Table 1: Test results.

Font Type	Title extraction	Subtitle extraction
AL-Quds	98.18 %	97.96 %
AxTManal	98.15 %	98.23 %
Beirut	/	98.87 %
AL-Quds Bold	98.45 %	98.17 %
Kacstone	97.56 %	97.55 %
Alshrek Titles	97.78 %	/
Total	98.02 %	98.15 %

Due to the absence of previous articles with the same goal in Arabic documents, we evaluated the performance of our approach by comparing it with related articles that have similar goals such as line segmentation. In [Mun17], they took a binarized image as input and the algorithm returned a data file that contains a segmented image. Though it went well (99%) for line segmentation with different fonts, it cannot be applied to a complex structure, due to a dependence on the vertical projection in the first phase of page segmentation, where there must always be a vertical white space on the whole page between the articles. Therefore there are incorrectly segmented lines with poor detection in the case of the absence of vertical white space in the page image. In [Abu06] the authors proposed a robust method for line segmentation and they score of 97.8% for both simplified and traditional text fonts (97.3% for simplified font and 98.4% for a traditional font), based on splitting one region into many smaller regions in a repetitive way until no more regions require splitting using a horizontal projection and a set of constraints. However, as the program does not work with a complex structure that has more than one article and different sizes of text on the same page, it is not possible to extract the lines for both normal text or large size text from each article if it exceeds an article on the image page or if it contains variable font size texts in the same article. Another study [Sou10] focused on the line segmentation of low-quality documents, by investigating different text-line segmentation algorithms like Projection Profiles (PP), the Run Length Smearing Algorithm (RLSA) and Adaptive Run-Length Smearing Algorithm (ARLSA); and by applying HPP they achieved a score of 100% on English documents that had varying spaces. RLSA achieved an accuracy of 96% on overlapping documents, and ARLSA achieved an accuracy of 99% on English documents with overlapping components. However, PP cannot handle images where the text lines are overlapping or touching. RLSA and ARLSA fail if there is any overlap between two text lines.

Figure 7: Samples of images used for the line segmentation method: (a) Ibrahim's data [Abu06]: an article with the same text size in one column; (b) Soujanya et al.'s data [Sou10]: an article with a different size font in one column; (c) Ayesh et al.'s data [Mun17]: variability of font size and the possibility of multiple articles, which was restricted by the presence of vertical white spaces between them; (d) Our own data where several articles have different font sizes and figures.

Although the program can handle many size fonts, in the previous study they based it on documents which had just one article hence one title had no more than this, and their approach cannot be applied on pages with a complex structure e.g. when there are many articles, figures, and titles. In [Bou18], they extracted the lines based on horizontal projection, local minima, and conflict resolution and got a score of 99.85% for text. They based it on more than one hundred pages taken from the same newspaper, and they had a similar structure for all the images they had for their dataset. Their text was simple text or titles because they did not distinguish between them, and this made it difficult to compare because a good segmentation line does not mean good title and subtitle detection. In fact, every article has a title, which leads us to think that good article detection with good line segmentation means a good title and subtitle extraction. In their study, they got a score of 90.03% for

ISBN 978-80-86943-38-1

article detection and for title detection they were unable to exceed this even in the best cases.

5 CONCLUSIONS

Title segmentation plays a significant role in the segmentation phase for the identification of any article in any random document paper. In this study, we handled the problem of distinguishing text and overlapping-lines with small font size, and for large fonts, using RLSA, Connected Components (CCs) and Projection Profile (PP) in scanned pages. We evaluated the proposed method on the real three hundred text images using the PATD database. The results presented here are superior to those of existing algorithms that perform the same task. Our future goal is to extend our document language procedure to other document formats and structures and generalize its capabilities.

6 REFERENCES

[Azo95] A. S. Azokly, "A Uniform Approach to the Recognition of the Physical Structure of Composite Documents Based on Spatial Analysis". PhD Thesis, Institute of Computer Science - University of Fribourg (Switzerland), 1995.

[Mon11] F. Montreuil, "Extraction of document structures by conditional random fields: application to handwritten mail processing". PhD thesis, University of Rouen, 2011.

[Gra09] A. Graves and J. Schmidhuber, Offline handwriting recognition with multidimensional recurrent neural networks, in Advances in neural information processing systems, pp.545-552, 2009.

[Ray15] A. Ray, S. Rajeswar, and S. Chaudhury, Text recognition using Deep BLSTM N, in Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on. IEEE, pp.1-6, 2015.

[Ryu14] Ryu, H. I. Koo, and N. I. Cho, Language independent text-line extraction algorithm for handwritten documents, IEEE Signal processing letters, Vol.21, No.9, pp.1115-1119, 2014.

[Buk08] S. S. Bukhari, F. Shafait, and T. M. Breuel, Segmentation of curled text lines using active contours, in Document Analysis Systems, 2008. DAS 08. The Eighth IAPR International Workshop on. IEEE, pp.270-277, 2008.

[Zek11] A. M. Zeki, M. S. Zakaria, and C.-Y. Liong, Segmentation of Arabic characters: A comprehensive survey, International Journal of Technology Diffusion, Vol.2, No.4, pp.48-82, 2011.

[Che01] A. Cheung, M. Bennamoun, and N. W. Bergmann, An Arabic optical character recognition system using recognition based segmentation, Pattern recognition, Vol.34, No.2, pp.215-233, 2001.

[Sou10] P. Soujanya, V. K. Koppula, K. Gaddam, and P. Sruthi, Comparative study of text line segmentation algorithmson low quality documents, CMR College of Engineering and Technology Cognizant Technologies, Hyderabad, India, 2010.

[Hus15] S. Hussain, S. Ali et al., Nastalique segmentation-based approach for Urdu OCR, International Journal on Document Analysis and Recognition (IJDAR), Vol.18, No.4, pp.357-374, 2015.

[Alj12] I. Aljarrah, O. Al-Khaleel, K. Mhaidat, M. Alrefai, A. Alzu bi, M. Rababah et al., Automated system for Arabic optical character recognition with lookup dictionary, Journal of Emerging Technologies in Web Intelligence, Vol.4, No.4, pp.362-370, 2012.

[Bro13] D. Brodic and Z. N. Milivojevi, Text line segmentation with the algorithm based on the oriented anisotropic Gaussian kernel, Journal of Electrical Engineering, Vol.64, No.4, pp.238-243, 2013.

[Bou18] H.Bouressace and J.Csirik, Recognition of the logical structure of Arabic newspaper pages, 21st International Conference on Text, Speech and Dialogue, Lecture Notes in Artificial Intelligence (Springer), Vol.11107, No.3, pp.251-258, 2018.

[Mun17] Muna Ayesh, Khader Mohammad, and Aziz Qaroush, A Robust Line Segmentation Algorithm for Arabic Printed Text with Diacritics, IS and T International Symposium on Electronic Imaging, pp.42-47, 2017.

[Zek11] A. M. Zeki, M. S. Zakaria, and C.Y. Liong, Segmentation of Arabic characters: A comprehensive survey, International Journal of Technology Diffusion, Vol.2, No.4, pp.48-82, 2011.

[Gor97] L. O Gorman and R. Kasturi, Executive briefing: document image analysis, IEEE Computer Society Press, ISBN 0-8186-7802-X, 1997.

[Ari07] M. Arivazhagan, H. Srinivasan, and S. Srihari, A statistical approach to line segmentation in handwritten documents, International Society for Optics and Photonics, 2007.

[Abu06] I. Abuhaiba, Segmentation of discrete Arabic script document images, Al Azhar University, Vol.8, No.1810-6366, pp.85-108, 2006.

[Sim08] Simone Marinai, Introduction to Document Analysis and Recognition, Studies in Computational Intelligence (SCI), Vol.90, No.1-20, 2008.

[Bou19] H.Bouressace and J.Csirik, Printed Arabic Text Database for Automatic Recognition Systems, 5th International Conference on Computer and Technology Applications, pp.107-111, 2019.

 ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

CNN Approaches for Dorsal Hand Vein Based Identification

Szidónia Lefkovits
University of Medicine, Pharmacy, Sciences and Technology
Department of Computer Science
Tg. Mureș, Romania
szidonia.lefkovits@science.upm.ro

László Lefkovits
Sapientia Hungarian University of Transylvania
Department of Electrical Engineering
Tg. Mureș, Romania

László Szilágyi
Sapientia Hungarian University of Transylvania
Department of Electrical Engineering
Tg. Mureș, Romania

ABSTRACT

In this paper we present a dorsal hand vein recognition method based on convolutional neural networks (CNN). We implemented and compared two CNNs trained from end-to-end to the most important state-of-the-art deep learning architectures (AlexNet, VGG, ResNet and SqueezeNet). We applied the transfer learning and fine-tuning techniques for the purpose of dorsal hand vein-based identification. The experiments carried out studied the accuracy and training behaviour of these network architectures. The system was trained and evaluated on the best-known database in this field, the NCUT, which contains low resolution, low contrast images. Therefore, different pre-processing steps were required, leading us to investigate the influence of a series of image quality enhancement methods such as Gaussian smoothing, inhomogeneity correction, contrast limited adaptive histogram equalization, ordinal image encoding, and coarse vein segmentation based on geometrical considerations. The results show high recognition accuracy for almost every such CNN-based setup.

Keywords

Biometric identification, dorsal hand vein recognition, CNN architectures, transfer learning.

1 INTRODUCTION

The increasing number of smart devices, cloud computing and home automation have made people more conscious of security and privacy. They have realised the importance of protecting their personal data stored in LAN or on servers in WAN networks.

The credentials in access control systems must be a unique ID that a user knows or owns. Based on this the user is associated and authenticated and it is given access to the requested resource.

Traditional identification and authentication methods are gradually losing ground because they can be easily hacked using a man-in-the-middle attack. The usual credentials such as tokens, passwords or ID-cards are not sufficiently reliable, since they can be relatively easily intercepted or falsified. Tokens, passwords and other important data may be caught or lost, and reused afterwards serving as fake documents. Finally, knowledge-based information may be easily forgotten. Biometrics is one of the most secure and reliable means of personal identification or authentication methods. Biometrics makes use of unique, unforgeable characteristics based on body measurements or comportment that are difficult to copy or steal.

The most important features used in biometric identification are the face, fiducial points, fingerprints, veins, palm and dorsal hand veins, finger veins, the iris, the retina, ears, the voice and the DNA.

In recent years, biometrics based on patterns of the vascular system have gained increasing attention. Analysis of finger vein, palm vein and dorsal hand vein patterns are the most common vascular structure biometrics. Their advantages are uniqueness, stability, contactless acquisition and unforgeability. They are also typical of the living.

In this paper, a dorsal hand vein identification system is presented which exploits the strength of convolutional neural network architectures.

The rest of the paper is organized as follows: after a short review of dorsal hand vein detection and authentication systems in the literature (Section 2), the database used in the experimental setup is presented (Section 3). Section 4 summarises the fine-tuned and retrained CNN architectures, followed by the detailed description of the approach proposed with two CNNs trained end-to-end (Section 5).

Finally, our experiments and results are presented in Section 6, followed by the conclusion and discussion (Section 7).

2 RELATED WORK

The blood circulation system of every creature is already formed in embryo state. The exact path and form of the veins, from the medical point of view, is not known in detail, and neither is the reason of the uniqueness of the vessel network. It is only known that the probability of finding two individuals with the same pattern of blood circulation system is quite low.

Technically, dorsal hand vein detection is the easiest compared to finger vein, palm vein or other vein pattern acquisitions. The veins are under the skin and carry blood from the organs towards the heart. They are bluish in colour and can be detected by light reflecting the temperature difference between the warm blood flow and the surrounding cells. The acquisition of dorsal hand images is usually done by Near Infrared (NIR) or Far Infrared (FIR) cameras. The acquisition protocol is not standardized and differs from one database to the other. The resolution of the images obtained is usually very low, with low contrast and a restricted region of grayscale intensities.

Vein-based approaches can be classified into two different types: shape-based methods and texture-based methods.

Shape-based methods use vascular structure information, extracting line or curve features and measuring different types of distances: the Hausdorff distance or the Line Segment Hausdorff distance or angles [1]. Here the geometric representation is obtained based on an approximation of segments by short vectors, bifurcation, endpoints and crossing points. These features are obtained after the vessel is detected and the skeleton of the vein structure is obtained. The segments of the skeleton can be approximated by line detection using Hough transform. In this case the veins are extracted by line tracking methods or local curvature extraction [2]. The disadvantage of these methods is the requirement for an aligned and registered skeleton structure.

The registration and alignment of two binary skeleton images is usually done by considering local invariant feature points such as SIFT (Scale Invariant Feature Transform) [3], SURF (Speed-Up Robust Features) [4] and Hessian-Laplace interest points [5].

Another approach capable of registering two vein skeleton images using a 3D rigid transformation that maximised the skeleton overlap was described in our previous article [6].

The texture-based methods can be divided into two categories: global texture methods and local texture methods. The global methods, also called holistic methods, extract the features from the whole region of interest, characterising the texture and vein structure globally. For this purpose, Principal Component Analysis (PCA) was used by Wang et al. in [7], the Fisher Linear Discriminant was applied Liu et. al in [8] and Independent Component Analysis (ICA) by Yuksel et al. [9]. These methods rely on eigenvalue and eigenvector decomposition. The results in these cases were severely influenced by the variation of position, viewpoint, contrast, illumination changes, occlusion, distortion and misalignment of images. However, these methods are a good starting point in combination with local texture methods.

The most important local texture-based approaches are different variants of locally binary patterns (LBP) or circular locally binary patterns (CLBP) [10] or Gabor texture descriptors [11]. Another type of local information extraction is based on local key point matching, for example SIFT (Scale Invariant Feature Transform) [12], Difference of Gaussian [13], Oriented Gradient Maps [14], Centroid-based Circular Key-point Grid (CCKG) [12].

The results obtained by a single key-point extraction method are not good enough; therefore, Huang et. al [15] proposed a combination of LBP for local texture, binary coding for local shape extraction and graph matching for global shape characterisation.

The most recent research field of convolutional neural networks is not widespread in the case of vein-based identification. Hong et al. proposed a reduced complexity four-layered CNN for finger vein recognition [16] and Wan et al. [17] proposed a CNN based on VGG-19 [18] for dorsal hand vein recognition.

3 THE DATABASE

The proposed approach is evaluated on the NCUT (North China University of Technology) Part A [8] dorsal hand vein dataset. It is the largest publicly available database used for automatic dorsal hand vein image recognition systems. The NCUT database contains 2040 near infrared images of dorsal hands of 102 individuals, and both hands have 10 samples each. Due to the low-cost acquisition equipment, this database contains low-quality images with a high noise level. All images were acquired by the same dorsal hand scanner, resulting in roughly aligned images. There are only small changes in translation, rotation and illumination between the images. We can observe more significant changes in viewpoint because of the hand rotating around the handle of the scanner, considered to be the *Ox* axis of the images. The NIR intensity is stable, but there is a circular variation in illumination from the centre to the margins because of a light source illuminating form

 ISBN 978-80-86943-38-1

above. Thus, some veins near the margins are difficult to distinguish, and vein pixel intensities are sometimes similar to skin pixel intensities. The acquisition equipment through the NIR camera determined the low contrast with a resolution of 640 ×480 pixels on an 8-bit greyscale image. Moreover, the dorsal hand occupies only about half of the available area and there are only 80 integer intensities, which restricts the range of contrast into the [101, 180] interval.

4 TRANSFER LEARNING

Transfer learning is used to fine-tune the weights of a given CNN architecture to be suitable for new input data. It has been shown that architectures based on transfer learning obtain better results in accuracy and performance than retraining the same network structure from the beginning.

In our experiments we created two networks built and trained from scratch, and fine-tuned four other well-known CNN architectures (AlexNet, VGG-16, ResNet and SqueezeNet) with pretrained weights.

4.1 Alex Net

AlexNet [19] was the winning architecture of the ImageNet [20] Challenge in 2012. It was the first architecture implementing parallelism in training the network. The original architecture has an input image size of 224×224 and 3-channel RGB image. It consists of 5 convolutional layers, 3 overlapping max-pool layers and two fully-connected layers at the end. The activation functions in the conv-layers was the ReLU. The first conv-layer has a depth of 96 with a kernel size of 11×11 and a stride of 4. The output of this layer is a 55×55×96 response, which is reduced to 27×27×96 by an overlapping max-pool layer of 3×3 kernels and stride=2. The second conv-layer has a kernel size of 5×5, a depth of 256, a stride of 1 and a zero-padding of 2 pixels in each direction. LRN (Local Response Normalization) was applied to the responses of the previous convolutional layers, i.e., batch-norm was not known at that time. The output is resized to half in width and height to 13×13×256 with the same type of max-pool as before. The following 3 convolutions have a kernel size of 3×3 with a padding of 1, astride of 1 and a depth of 384, 384 and 256, respectively. These 3 layers maintain the previous size of 13×13×256, which is reduced to half 6×6×256=9216 afterwards. These responses are fed into 2 FC layers, each having 4096 activations. The final decision of predicting 1000 class layers is made by a softmax function.

In our approach we refined the trained weights for ImageNet and replaced the final FC layer to be suitable for our problem of dorsal hand vein identification by changing the output layer to 204 instead of 1000.

4.2 VGG

VGG was introduced by Simonyan and Zisserman [18] and was the winning architecture of the ImageNet competition in 2014. It is a much deeper model compared to AlexNet, containing 16 (VGG-16) and 19 (VGG-19) layers, respectively. Their idea was to stack multiple convolutional layers, with each having a kernel size of 3×3 with stride 1. This kernel size reduced the number of parameters in the layer. It has been shown that two stacked 3×3 kernel layers have the same receptive fields as a single 5×5 one. Likewise, a stack of 3 conv-layers has the same receptive field as a single 7×7 kernel conv-layer. The VGG-16 has 13 conv-layers, 5 max-pooling layers and 3 fully connected layers at the end. The other novelty here was the doubling of filter depth after reducing the input image size to half in both height and width. This led to a parameter reduction of only 1/2 instead of 1/4. The input image size of 224×224×3 was reduced after two conv-layers with a depth of 64 (conv1_1, conv1_2) to 1/2 size using a max-pool layer with stride 2. , Two other conv-layers were applied on the 1/2 image size, with double depth 128 and a max-pooling resize of the output to 1/4. Three conv-layers were stacked on the 1/4 size, each having a depth of 256. On the 1/8 and 1/16 sizes, 3-3 stacked convolutional layers followed, each having a depth of 512. The output of the last layer, in this way, was 7×7×512, and was fed into three FC layers of sizes 4096, 4096 and 1000. The last two FC layers were followed by dropout with a probability of 0.5, like in AlexNet. This architecture has about 138 million parameters and was extremely hard to train. The training was done in several stages, gradually adding the interior layers.

In our approach we refined the initially trained weights for ImageNet and replaced the final FC layer in order to obtain 204 class scores.

4.3 ResNet

The main reason for introducing ResNet [21] was the observation of lower training performances, although the number of layers increased. Thus, CNNs with a high number of convolutional layers are more difficult to be optimized. Moreover, if the number of layers is above a certain limit between 20 and 30, the CNN will also have a higher training and test error. The convergence of such deep architectures is not reached. The authors of paper [21] solve the problem of mapping an input layer to an output layer by computing only the difference that must be added (subtracted in case negative values) to the input values. So, instead of determining *Out(x)* it determines the residual *R(x)*, where *R(x)=Out(x)-x*. The shortcut connection, also called bypass, is the identity mapping (*x*) which adds the input to the residual and obtains the output. The residual is

ISBN 978-80-86943-38-1

determined by multiple (2 or 3) consecutive conv-layers. In this way they are able introduce very large networks with 18, 34, 50, 101 and 152 layers. Every conv-layer is batch normalised and fed into ReLU activation.

In our experiments we have used the ResNet50 architecture. The input layer is the dimension used in ImageNet. The first layer is 7×7×64 with stride 2. The next dimension is 112×112 because of resizing by max-pooling. On this size, 3 consecutive conv-layers are used to determine the residual (1×1×64, 3×3×64, 1×1×256), and these are repeated 3 times. The next feature map dimension is 56×56. Here the residual is computed using another 3 conv-layers (1×1×128, 3×3×128, 1×1×512), and these are repeated 4 times. The next input feature dimension is 28×28, and here the residual block consists of these three 1×1×256, 3×3×256, 1×1×1024 conv-layers. These layers are replicated 6 times. On the final dimension of 14×14, the 3 layers are included in the residual block (1×1×512, 3×3×512,1×1×2048). The last layer is a fully-connected layer of 7×7×2048, mapping to 1000 class scores determined by softmax.

In our experiments we have fine-tuned the pretrained weights of this architecture using the NCUT dataset, added data augmentation and adapted the weights and FC layers according to the dorsal hand identification problem.

4.4 Squeeze Net

The SqueeezeNet architecture introduced in [22] is a CNN with a drastically reduced number of parameters (of 1.2 million only), but with accuracy results in ImageNet object detection completion similar to AlexNet. Because of the lower number of parameters, it requires less memory in parallel training, and the model obtained can be easily loaded onto embedded systems or microchip-based intelligent systems with lower hardware resources. The main ideas of this network are the so-called fire modules along with the conservation of the activation maps by downsampling their spatial dimension later and fewer times in the network. The fire module consists of 3 components: a squeeze layer (of kernel 1×1) to reduce the depth of the input filter-map (reducing the number of parameters) and two expansion layers of kernel 1×1 and 3×3 to transform the reduced layer back to its original size. $s_{1\times1}$ squeeze-layers have the role of reducing its input size of $w\times h\times d$ to only $w\times h\times s_{1\times1}$. The squeeze layers are followed by ReLU activation and two consecutive expansion layers: the 1×1 conv-based returns to a higher dimensional space $w\times h\times e_{1\times1}$ and the 3×3 conv-based reduces the number of weights compared to the previous layer. After the expansion the output layer is of size $w\times h\times (e_{1\times1}+e_{3\times3})$ and is followed by a ReLU activation. The SqeezeNet architecture is made up of a conv-layer, max-pool for downsampling, three fire modules, max-pool again, four fire modules, max-pool, a final conv-layer and average pooling at the end, meaning a total of 11 layers based on convolutions. The model trains faster not only because of the fire modules, but also due to using sparse weight matrixes and computation not with float numbers but 6- or 8-bit quantized integers [23]. The authors present three variants of the SqueeezeNet: the vanilla architecture, the SqueezeNet with bypass connections and bypass with convolutions.

In our application we used the vanilla SqueeezeNet without bypasses retrained the original network and adapted the last FC layer to the purpose of dorsal hand vein identification. Thus, instead of 1000 class scores we had to obtain only 204.

5 OUR APPROACH

The goal of this research is the evaluation of two novel CNN network architectures proposed and trained for dorsal hand vein recognition. Their identification accuracy and training and testing performances are compared to well-known CNN architectures such as AlexNet, VGG-16, ResNet-50 and SqueezeNet, summarised above.

The first network proposed and trained from scratch is a 6-layered CNN made up of 4 convolutional and 2 fully connected layers. The input image was redimensioned and centre-cropped to the same dimension as required for the other types of networks, i.e., to 224×224×3. The original image is only grayscale, but the depth of 1 was converted by replication to a depth of 3.

The NCUT dataset has only 10 different image samples of both hands for every subject. This number of training images is very small for proper end-to-end training of convolutional neural networks

In order to enlarge the training dataset available, we have applied classic data augmentation methods to generate, instead of 10, thousand similar images during the training process. There are several types of augmentation techniques. We have used centre-crop based on a binary mask to consider only the hand, followed by a resize to the standard input size of all networks. We applied random rotation of the images by ±10°. Finally, the images were normalized with Z-score normalisation using the mean and standard deviation of the image set.

In this study, our goal was to create a simple CNN model which is easily trained and has only a few parameters, but obtains reasonable identification accuracy compared to other state-of-the-art pretrained CNNs with initial weights fine-tuned for the ImageNet Challenge, these are much harder to be trained.

 ISBN 978-80-86943-38-1

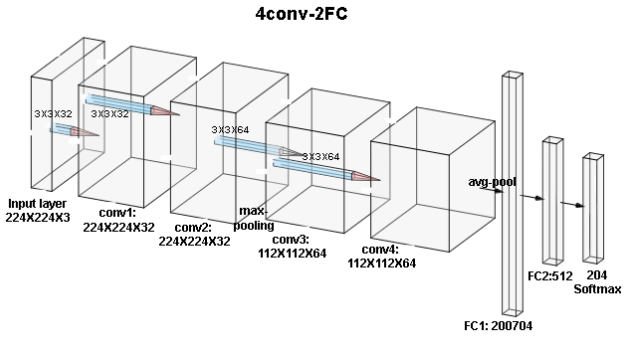


Figure 1: 4conv-2FC Architecture Trained from Scratch

	Layer type	No. of filters	Size of input feature map	Size of output feature map	Size of kernel	No. of stride	No. of padding
Dimension 224×224	Conv1_1	32	224×224×3	224×224×32	3×3	1×1	1×1
	Conv1_2	32	224×224×32	224×224×32	3×3	1×1	1×1
	Batch norm 4×32						
	Max-pool 1		224×224×32	112×112×32		2×2	0×0
	Dropout p=0.25						
Dimension 112×112	Conv2_1	64	112×112×32	112×112×64	3×3	1×1	1×1
	Conv2_2	64	112×112×64	112×112×64	3×3	1×1	1×1
	Batch norm 4×64						
	Avg-pool 2		112×112×64	56×56×64		2×2	0×0
	Dropout p=0.25						
	FC1		56×56×64	512			
	Dropout p=0.25						
	FC2		512	204			

Table 1: 4conv-2FC architecture

To this end, we have first built a CNN with 4 convolutional layers, called 4conv-2FC (Figure 1). The convolutional filter in each case was a 3×3 convolution. The 3×3 filter is a reasonable dimension to extract a patch of the image. If the filter is larger, the number of weights between two layers will be higher, implying the growth of the final number of parameters. If the input has a depth of D and the filter $f \times f \times d_f$, the number of weights between the particular input and the filters output will be $(w_f \times h_f \times d_f + 1) \times D$. The 1 constant represents the bias term. In the literature [18], it has been shown that a 5×5 kernel response can be obtained as a stack of two conv-layers with a kernel of 3×3. It is obvious that $2\times(3\times3\times d+1)\times D$ is 40% less than $(5\times5\times d_f+1)\times D$. Every convolutional layer output is fed into a ReLU activation function. The rectifying linear unit $ReLU(x)=max(0,x)$ is preferred instead of other activations because the its derivate is easier to compute. The hyperbolic tangent and sigmoid activations are rarely used in CNNs because they quickly saturate, and their derivate in that case is 0.

Every conv-layer is normalized using batch normalization. Batch normalization (BN) learns the mean and standard deviation of a given layer and preserves them even if the input weights are changing. It makes the learning of subsequent layers easier and has a regularisation effect as well. For every depth (d), the BN learns the γ (scale) and (β) shift parameters of the normalized outputs of a given layer. For the normalisation of the values, it computes the mean and standard deviation of the layer; therefore, the number of parameters in this case is $4 \times d$. We use a dropout with a probability of 0.25 and 0.5 after the second and fourth conv-layer. The convolutional layers maintain the same feature map size in width and height by using a zero-padding of p=1 and stride s=1 for every 3×3 convolutional kernel. Only the depth of the feature maps changes. The output width (w_O) and height (h_O) can be calculated based on the input feature maps size $w_I \times h_I$, the filter size $w_f \times h_f$ and the padding p and stride s, according to the formula: $w_O=((w_I-f_I)+2*p)/s+1$.

ISBN 978-80-86943-38-1

The depths of the feature maps gradually increase by a factor of 2 when their width and height is reduced by 2. This operation reduces the number of activations between layers by a factor of only 2 instead of a factor of 4. The smaller size of the feature maps is obtained by max-pooling layers of 2×2 with a stride of 2. This means that the feature map is reduced to half its size by keeping the maximum value in every block of 2×2 pixels. The last pooling layer in our network is global average pooling in blocks of 2×2, with the role of minimising the overfitting before the first dense layer. The convolutional layers are followed by 2 fully connected layers. The last fully connected layer is fed into a softmax function. The softmax converts the class scores obtained into probability distributions, representing the probability of the input image to be considered in class *i*, where *i* is the number of subjects in the database ×2 (left/right land), in our case 2×102=204.

Our first CNN architecture 4conv-2FC (Figure 1, Table 1) considers two dimensions of the original image: the full size (224×244 feature map) and the half-size feature map (112×112). For each dimension, two convolutional layers are considered. The first layer has an input of 224×244×3 and produces an output of 224×244×32 followed by another similar layer. These layers are obtained using 3×3×32 filters with s=1 and p=1. After these two layers, the feature map is reduced to half its size, and two conv-layers are computed again. The third has an input of 112×112×32 and an output of 112×112×64 fed into the forth, obtaining an output of 112×112×64. Before the dense layers, the output is again reduced to 56×56×64 (=200704). Thus, the first fully connected layer creates a mapping between 200704 and 512. The last FC layer has a number of outputs equal to the number of classes, i.e., 204. The parameters of this CNN architecture are summarized in Table 1.

Our second CNN architecture 6conv-2FC (Figure 2, Table 2) considers the dimensions of the original image: the full size, the half-size feature map (112×112) and the 1/4 size one (56×56). For each dimension, two convolutional layers are added. The first four conv-layers are similar to the first approach, but there is another pair of conv-layers at the 56×56 size. The fifth layer has an input of 56×56×64 and obtains an output of 56×56×128, while the sixth layer is a replica of this, obtaining once more an output of 56×56×128.

Before the dense layers, the output is again reduced to 28×28×128 (100352). Thus, the first fully connected layer makes a mapping between 100352 and 1000. The last FC layer must have 204 responses, considering the number of classes. The parameters of this CNN architecture are summarised in Table 2.

6 RESULTS AND EXPERIMENTS

In this section we describe our experiments related to dorsal hand vein recognition with different convolutional network architectures. We compare the effect of several preprocessing steps on the recognition performance of the trained networks.

The databased used in our approach is the NCUT database described in detail in Section 3.

The block diagram of our system is depicted in Figure 3. The flowchart of our approach has the database as the input, followed by a ROI detection, based on a binary mask, different preprocessing steps, and Z-score normalization with a mean of 0 and a standard deviation of 1. The main part of the system is the implemented and trained CNN architecture used for dorsal hand vein identification purposes.

The hardware used for carrying out our experiments was a Windows 10 64bit operating system with 32GB of memory and an NVIDIA Geforce GTX Graphics Card with 11 GB of memory using the Python and Pytorch framework.

In our experiments we considered different types of settings. In the first settings we compared our proposed two different CNN architectures trained from scratch. These two architectures considered the same original dataset. We have split the dataset randomly into 6 samples per person for training and 4 for testing and out of these 2 for validation. In both cases, we considered the same number of training epochs (400) and the same hyperparameters (optimization of Stochastic Gradient Descent with a learning rate of 0.01).

The number of parameters in these two cases are approximately the same 103M (4conv-2FC) or 100M (6conv-2FC). The training and the validation curves in Figure 4 show a better dropdown of the training curve and a better stabilization of the validation curves for the CNN architecture with 6 conv-layers. The accuracy of the networks is 92% and 95%. (Figure 9 and Table 3).

In the next stage, we compared the training and testing performance versus the number of epochs, the learning rate and the optimization criterion only on 6conv-2FC.

A total number of iterations of about 400-500 was enough in the case of the two networks trained from scratch, and only 200-250 iterations run were enough for the pretrained and fine-tuned networks.

A low learning rate of 0.001 led to a very slow convergence and a higher learning rate of 0.1 showed an oscillating behaviour in the training loss.

 ISBN 978-80-86943-38-1

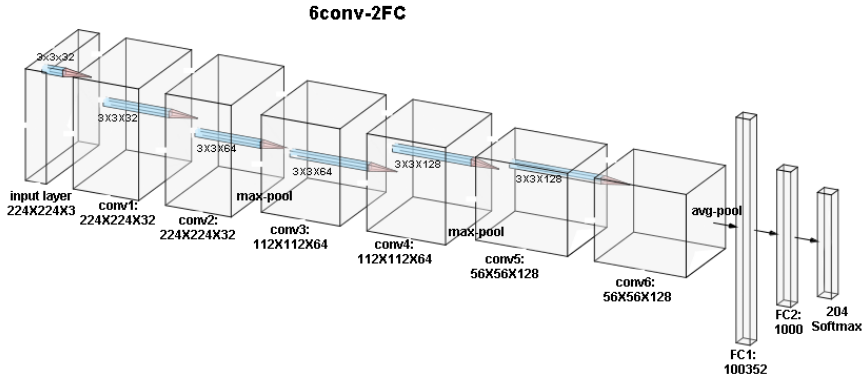


Figure 2: 6conv-2FC Architecture Trained from Scratch

	Layer type	No. of filters	Size of input feature map	Size of output feature map	Size of kernel	No. of stride	No. of padding
Dimension 224×224	Conv1_1	32	224×224×3	224×224×32	3×3	1×1	1×1
	Conv1_2	32	224×224×32	224×224×32	3×3	1×1	1×1
	Batch norm 4×32						
	Max-pool 1		224×224×32	112×112×32		2×2	0×0
	Dropout p=0.25						
Dimension 112×112	Conv2_1	64	112×112×32	112×112×64	3×3	1×1	1×1
	Conv2_2	64	112×112×64	112×112×64	3×3	1×1	1×1
	Batch norm 4×64						
	Max-pool 2		112×112×64	56×56×64		2×2	0×0
	Dropout p=0.25						
Dimension 56×56	Conv3_1	128	56×56×64	56×56×128	3×3	1×1	1×1
	Conv3_2	128	56×56×128	56×56×128	3×3	1×1	1×1
	Batch norm 4×128						
	Avg-pool 3		56×56×128	28×28×128		2×2	0×0
	Dropout p=0.25						
	FC1		28×28×128	1000			
	Dropout p=0.5						
	FC2		1000	204			

Table 3: 6conv-2FC architecture

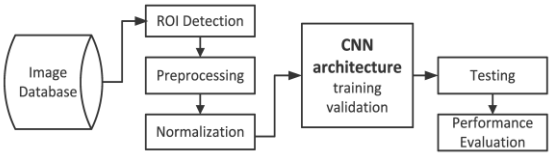


Figure 3: The Proposed System

Thus, the most adequate value for the learning rate in our case was 0.01. We also compared two different weight update methods, simple SGD with lr=0.01 as well as SGD with momentum (lr=0.01, β=0.9) and RMSProp updates. RMSProp update led to an oscillating behaviour, and simple SGD worked slightly better when training the two new networks; at the same time, SGD with momentum worked well in transfer learning.

We also proposed to compare different types of known networks and fine-tuned them by modifying their weights and final decision layer tailored to the problem of dorsal hand vein identification. Here we compared the training and test performances of the state-of-the-art networks described in Section 4.

Figure 5 compares the training behaviour. Here we can observe that the ResNet-50 network obtains the best training loss (with 50 conv+FC-layers), our 6conv-2FC (6+2 layers) and 4conv-2FC (4+2 layers) network is slightly worse, but their performance is similar to VGG-16 (13+3 layers) in this case.

ISBN 978-80-86943-38-1

The training loss of these four networks drops from the early epochs. AlexNet and SqueezeNet training losses drop in a relatively late epoch compared to the other four networks mentioned above.

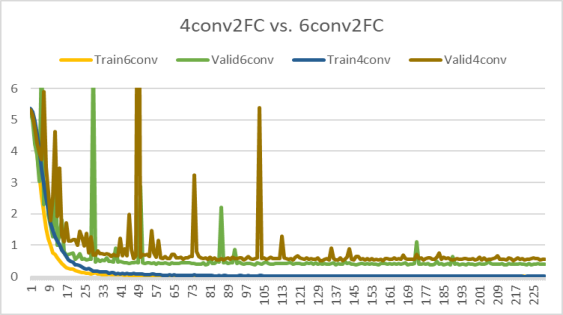


Figure 4: Training and Validation Losses for 4conv-2FC and 6conv-2FC

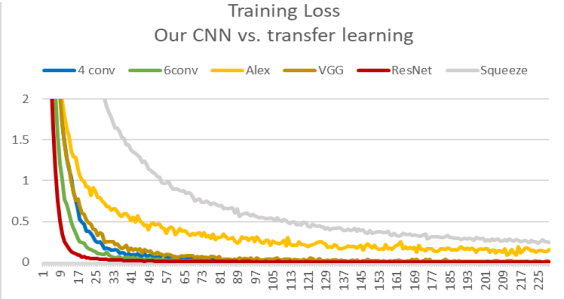


Figure 5: Training Loss in Transfer Learning

The 3×3 filters computed in the case of 6conv-2FC are shown in Figure 6. In fact, the kernel filters represent the most common 3×3 image patches in a given layer.

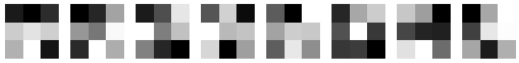
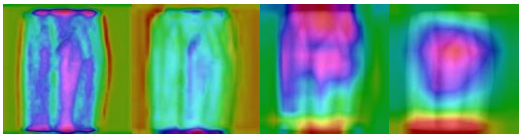


Figure 6: 3×3 conv-filters

The layer-wise heatmap obtained from GradCam [24] visualizations shows (Figure 7) the most activated regions of the dorsal hand layer by layer, from layer 1 to 4.



a) Layer1 b) Layer2 c) Layer3 d) Layer4

Figure 7: Heatmap of Layers 1-4

The activation is gradually concentrating on the upper part of the hand, where the main longitudinal veins fork into diagonal veins going into the fingers. This is the most representative part of the hand vessel system from the perspective of identification.

Our next experiment analysed the effect of the various preprocessing steps applied to the original image before the training of different CNN architectures. In our previous paper [6] we proposed a geometry-based vein extraction algorithm.

The process started with some image preprocessing steps followed by segmentation, which allowed extracting the skeleton of the detected veins. These preprocessing steps are image inhomogeneity correction, contrast limited adaptive histogram equalization and codification of the veins with an ordinal measure.

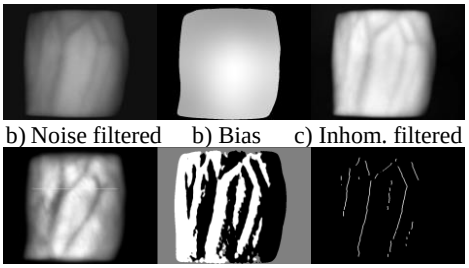
The preprocessing steps mentioned (Figure 3) are applied using a binary mask representing the ROI of the hand. Its boundary is determined by the contour of the black background and the non-black foreground. The next step (a) is the noise filtering (Figure 8a) of the masked image. Here we used a 2D Gaussian kernel of 3×3 with $\mu=0$ and $\sigma=0.5$. This eliminated the blurriness of the images. In addition, we noticed that the illumination of the images is not uniform. The intensity of the central part is much higher than at the margins. An inhomogeneity correction filter can reduce the variation in illumination. The variation in intensities can be modelled using a bias image (step b) (Figure 8b), which is a smooth multiplicative field over the filtered image. Thus, we solved this artefact by adapting the N3 inhomogeneity correction (step c) algorithm [25] to the given images (Figure 8c). Another important step (d) is contrast limited adaptive histogram equalization (CLAHE) (Figure 8d), which ensures the image quality and contrast enhancement required for better segmentation or identification. The parameters used in this case are a block size of 7 and 127 for the contrast-limiting threshold.

Finally, the images are segmented based on an ordinal image encoding (step e) procedure by comparing the successive pixel intensities along a horizontal line. Pixels with decreasing intensities along the line are coded black (0), pixels with increasing intensities are coded white (255), and pixels with almost constant intensities are coded grey (128) (Figure 8e). This encoding creates images that are a sort of relief-like representation of the veins. The threshold (T) which determines increasing, decreasing or constant behaviour is a parameter imposed by the width of the veins detected. A value of 0.5 for this threshold and a minimum region around the minimum point larger than 4 pixels is considered to be a vein centre. By applying this segmentation procedure (step f) we obtain coarse vein segmentations (Figure 8f).

The presented CNN networks were trained and tested on the original images, on the inhomogeneity-corrected and CLAHE images, on the coded and on the segmented images as well. The results for each type of image are summarized in Table 3 and Figure

ISBN 978-80-86943-38-1

9. We can observe that the best identification results are obtained on preprocessed images with steps a, b, c and d (column InhCLAHE).



b) Noise filtered b) Bias c) Inhom. filtered

d) CLAHE e) Coded f) Segmented

Figure 8: Preprocessed Images

The accuracy is 2-3% better than that for the normalized original images without applying inhomogeneity correction and CLAHE (column Orig.). The codification based on the ordinal measure did not bring any advantages to the training of the system or to testing accuracy. It is obvious from the results that the segmented black and white images with very few white (information) pixels and a lot of background (black) ones make the final identification performance much worse. This behaviour is because of the many background pixels contributing to a vanishing gradient most of the time.

Comparing the recognition accuracies of CNNs fine-tuned for the NCUT dataset and using the pretrained weights via transfer learning, we can observe that the ResNet-50 network has almost perfect 99% results for the test set, followed by VGG-16's 97-98%. AlexNet has similar results to our 6conv-2FC network, 95-96% and 94-95%, respectively. The worst recognition results were obtained by the SqueezeNet network. We can see that by combining any two networks as a decision ensemble and joining their softmax probability responses, 99-100% accuracy can be obtained.

CNN	Orig	InhCLAHE	Coded	Segment
4conv2FC	0.9195	0.9265	0.9142	0.8456
6conv2FC	0.9485	0.9583	0.9412	0.8775
alexnet	0.9650	0.9608	0.9510	0.9167
vgg	0.9722	0.9804	0.9706	0.8946
resnet	0.9951	0.9951	0.9951	0.9804
squeezenet	0.9069	0.9118	0.8824	0.7598

Table 3. Identification Accuracy on the Test Set

Our approach and results can be compared to the most recent state-of-the-art papers using CNN architectures. Paper [17] presents a variation of RCNN with self-growth strategy. They report a 95% recognition rate on the training and 91.25% on the test set. However, the database used in that case is unknown, i.e. not publicly available. Thus, the accuracy results are not referring to the same dataset.

The results of the other article [26] presenting dorsal hand vein recognition by CNNs is much more comparable to our experiments. They work with the same NCUT database and create a system with five parallel SqueezeNets. The accuracy obtained by majority voting of these parallel networks is 99.52%. This architecture which needs a five-times as much training and 5 times as many parameters, as our networks. Yet we demonstrated that by combining the outputs of any two of our trained networks a result of 99-100% can be easily achieved.

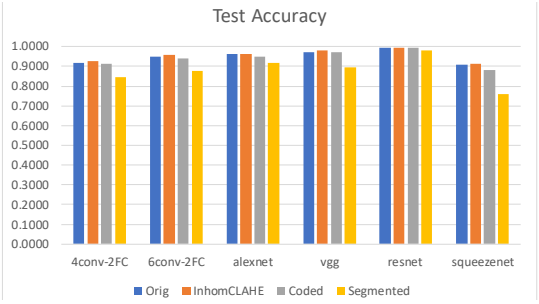


Figure 9: Identification Accuracy on Test Set

7 CONCLUSION AND DISCUSSION

In this paper, two different CNN architectures are proposed for dorsal hand vein-based identification. These CNNs are compared to existing state-of-the-art deep learning approaches. The training process based on the fine-tuning of the final layers and the parameter transfer on the rest of the pretrained weights lead to better identification performances, especially in the case of the VGG and ResNet architectures, compared to the end-to-end trained proposed CNNs (4conv-2FC and 6conv-2FC). The number of layers and the number of parameters for our CNNs is much smaller than in the other two better architectures. Therefore, they are more adequate in systems with lower hardware resources. However, the experiments show superior performance of deep learning training compared to other dorsal hand vein identification systems based on feature extraction.[12, 14, 15] In addition, we have found that different kinds of preprocessing steps, especially inhomogeneity correction of the images, gains a 2-3% increase in accuracy.

To generalise the identification process, a more complex database with much more subjects and image samples would be required for real-life validation and everyday use. Therefore, in the future we propose to enlarge the training dataset using several augmentation methods, and extend the NCUT database. Moreover, we intend to validate the proposed architecture not only for dorsal hand veins, but for palm veins and finger vein recognition also.

ISBN 978-80-86943-38-1

8 ACKNOWLEDGMENTS

The work of L. Lefkovits was partially supported by a grant from the Sapientia KPI, the research carried out was facilitated by Domus Hungarica Scholarship. L. Szilágyi is János Bolyai Fellow of the Hungarian Academy of Sciences.

9 REFERENCES

[1] M. M. Pal and R. W. Jasutkar, "Implementation of hand vein structure authentication based system," *Int. Conf. on Communication Systems and Network Technologies*, pp. 114–118, 2012.

[2] K. Chatfield, V. S. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: an evaluation of recent feature encoding methods.", *British Machine Vision Conf.*, vol. 2, no. 4, 2011

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Jour.of Computer Vision*, vol. 60: no.2, pp. 91–110, 2004.

[4] M. Pan and W. Kang, "Palm vein recognition based on three local invariant feature extraction algorithms," *Chinese Conference on Biometric Recognition*. Springer, pp.116–124, 2011.

[5] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," *Computer Vision ECCV 2002*, pp. 128–142, 2002.

[6] S. Lefkovits, S. Emerich, and L. Szilágyi, "Biometric system based on registration of dorsal hand vein configurations," *Pacific-Rim Symposium on Image and Video Technology*. Springer, pp. 17–29, 2017.

[7] L. Wang, G. Leedham, and D. S.-Y. Cho, "Minutiae feature analysis for infrared hand vein pattern biometrics," *Pattern recognition*, vol. 41, no. 3, pp. 920–929, 2008.

[8] Y. Wang, K. Li, and J. Cui, "Hand-dorsa vein recognition based on partition local binary pattern," in *Int. Conf. on Signal Processing (ICSP), 10th* . IEEE, pp. 1671–1674, 2010.

[9] A. Yuksel, L. Akarun, and B. Sankur, "Hand vein biometry based on geometry and appearance methods," *IET Computer Vision*, vol. 5, no. 6, pp. 398–406, 2011.

[10] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE PAMI* no.7, pp.971–987, 2002.

[11] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei, "Object-centric spatial pooling for image classification," *European Conference on Computer Vision*. Springer, pp. 1–15, 2012.

[12] D. Huang, R. Zhang, Y. Yin, Y. Wang, and Y. Wang, "Local feature approach to dorsal hand vein recognition by centroid-based circular key-point grid and fine-grained matching," *Image & Vision Computing*, vol. 58, pp. 266–277, 2017.

[13] Y. Wang, Y. Fan, W. Liao, K. Li, L.-K. Shark, and M. R. Varley, "Hand vein recognition based on multiple keypoints sets," in *IAPR Int. Conf. on Biometrics (ICB)*. IEEE, pp. 367–371, 2012.

[14] D. Huang, W. B. Soltana, M. Ardabilian, Y. Wang, and L. Chen, "Textured 3d face recognition using biological vision-based facial representation and optimized weighted sum fusion," in *CVPR*. IEEE, pp. 1–8, 2011.

[15] D. Huang, X. Zhu, Y. Wang, and D. Zhang, "Dorsal hand vein recognition via hierarchical combination of texture and shape clues," *Neurocomputing*, vol. 214, pp. 815–828, 2016.

[16] H. Hong, M. Lee,K. Park, "Convolutional neural network-based finger-vein recognition using NIR image sensors" *Sensors*,17:6, p. 1297, 2017.

[17] J. Wang and G. Wang, "Hand-dorsa vein recognition with structure growing guided cnn," *Optik*, vol. 149, pp. 469–477, 2017.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *preprint arXiv:1409.1556*, 2014.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks"*Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," 2009.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE CVPR*, pp. 770–778, 2016.

[22] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size," *preprint arXiv:1602.07360*, 2016.

[23] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *preprint arXiv:1510.00149*, 2015.

[24] R. R. Selvaraju, A. Das, R. Vedantam, et al., "Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization,"*CoRR*,abs/1610.02391, 2016.

[25] N. Tustison, B. Avants, P. Cook et. al, "N4itk: Improved n3 bias correction"*IEEE Trans Medical Imaging*,29:6,pp.1310–1320,2010.

[26] H. Wan, L. Chen, H. Song, and J. Yang, "Dorsal hand vein recognition based on convolutional neural networks. BIBM pp. 1215-1221, 2017.

 ISBN 978-80-86943-38-1

ISSN 2464-4617 (print) Computer Science Research Notes
ISSN 2464-4625 (DVD) CSRN 2901 WSCG Proceedings Part II

Improving facial attraction in video

Maycon Prado Rocha Silva
State University of Campinas
Brazil, Campinas, SP
mayconrochasilva@gmail.com

José Mario De Martino
State University of Campinas
Brazil, Campinas, SP
martino@fee.unicamp.br

ABSTRACT

The face plays an important role both socially and culturally and has been extensively studied especially in investigations on perception. It is accepted that an attractive face tends to draw and keep the attention of the observer for a longer time. Drawing and keeping the attention is an important issue that can be beneficial in a variety of applications, including advertising, journalism, and education. In this article, we present a fully automated process to improve the attractiveness of faces in images and video. Our approach automatically identifies points of interest on the face and measures the distances between them, fusing the use of classifiers searches the database of reference face images deemed to be attractive to identify the pattern of points of interest more adequate to improve the attractiveness. The modified points of interest are projected in real-time onto a three-dimensional face mesh to support the consistent transformation of the face in a video sequence. In addition to the geometric transformation, texture is also automatically smoothed through a smoothing mask and weighted sum of textures. The process as a whole enables the improving of attractiveness not only in images but also in videos in real time.

Keywords

Video processing, computer vision, artificial intelligence, image processing, attractiveness improvement, texture processing, geometric transformation.

1. INTRODUCTION

Beauty today is desired by many individuals. Past studies[Sla00][Lan87] demonstrate that people who are considered more attractive usually tend to be perceived as more competent and, as a consequence, to be more favored in life than the less attractive ones. A beautiful person is naturally more attractive and holds the attention of a spectator more closely, favoring, for example, advertising, journalism and also teaching. Industry has explored the trend looking for ways to highlight and improve the attractiveness of people. In this context, the face has a high relevance and consequently has been extensively worked on to become more attractive, both by definite means and by computational corrections.

The contributions given by the area of computer graphics have played an important role in assisting facial enhancement, removing and smoothing faults or imperfections. Sophisticated applications have been widely used to remove skin irregularities and measure.

Researches [Let15][Ley08][Eis06][Guo09] on improving facial attractiveness show results very close to the natural. The main forms are through geometric processing, overlapping textures and smoothing masks. In these researches, all the techniques presented use aided processing, which depends on point of interest marking, or on image positioning and/or specific equipment. This makes the use more limited to a studio and can be applicable only to static images.

This article presents an approach for improving facial attractiveness in videos, where, in real time, face identification, mapping of points of interest, identification of a similar individual, geometric processing and texture smoothing occurs.

2. RELATED WORK

Leite and De Martino [Let15] present a study to improve facial attractiveness in two-dimensional front images using geometric transformation and texture smoothing. The results validated in the research indicate success in raising the attractiveness of processed samples. The mentioned limitations are: manual marking of points of interest of the face, neutral expression only and manual identification of models.

Another work presents an application launched in 2016 by an autonomous group of

Russian developers the FaceApp [Lon17] (Figure 1). Capable of modifying images in a dynamic way, such as: changing expression, rejuvenating, adding props, changing gender, changing hair, among others.

Figure 1. Images generated using FaceApp

Similarly, the Meitu application [Con19] (Figure 2) features functions that add effects that make photos more attractive, such as: smoothing the skin, facial reduction, and makeup application.

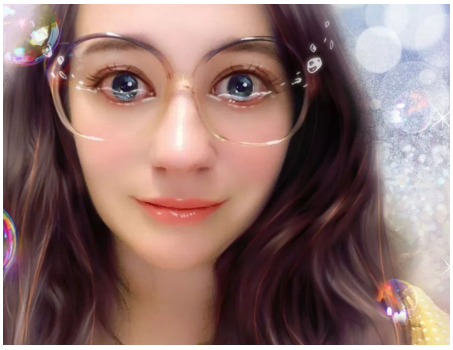


Figure 2. Image generated using Meitu

The developers of FaceApp and Meitu applications cite that machine learning and computer vision techniques have been used through free source libraries such as TensorFlow [Aba16] and OpenCV [Bran00].

The disadvantage of both FaceApp and Meitu is the application only in pre-defined models and the impossibility of extending the use in videos.

In 2017, using real-time video processing, Snapchat added in its mobile application a functionality capable of transforming face components into different configurations such as face change, zombie transformation and tracking of three-dimensional models that can be added according to the position of the face detected in the video.

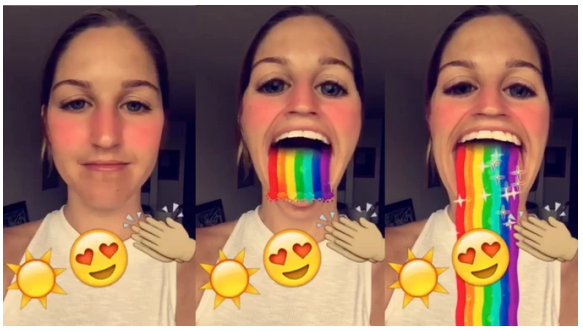


Figure 3. Images generated using Snapchat

Our work resembles the results of FaceApp and Snapchat [Cne17](Figure 3) applications in aspects of Face Detection, geometry transformation and processing of three-dimensional models . However, it is directed with the theme of improving attractiveness by adding techniques presented in Leite's [Let15] work. Expand than focus on video processing and automatic detection of point of interest.

[Fan2016] present a method using a default mesh based on the local chin shape that has great effect on facial attractiveness, and [Cho2012] replace facial features dynamically on videos, these two approaches, different this work, uses different model templates not based on common distances.

[XI2016] present the temporal-spatial-smooth warping (TSSW) method to achieve a high temporal coherence for video face editing. TSSW is based on two observations: (1) the control lattices are critical for generating warping surfaces and achieving the temporal coherence between consecutive video frames, and (2) the temporal coherence and spatial smoothness of the control lattices can be simultaneously and effectively

 ISBN 978-80-86943-38-1

preserved. The framework proposed can be used to improve the work presented in this article combined with the model similarity detection .

[Zha2017] and [Dia2019] combine different ways to make facial beautiful in static images with expressive results using models with no similar facial features distances.

3. PROPOSED SOLUTION

Tatiane and De Martino [Let15] presents a study to improve facial attractiveness in front images in two dimensions. The results were presented in obtaining a more attractive face exploring warping technique. However, some significant points are highlighted for continuity of new works, as follows:

- Automatic detection of points of interest.

- Projection of points of interest in three-dimensional coordinates.

- Video processing;

These points served as an inspiration for the work presented in this article, and the evolution made addresses continuity. That is, the results extend the technique presented by [Let15] in complement with the automatic detection of points in a static image and three-dimensional projection. The process covers the concepts of automatic detection of points of interest as well as the real-time identification of a basic model for transformation. The methodology is organized in stages, according to Figure 4.

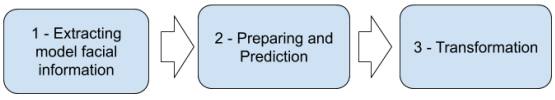


Figure 4. The stages of the method

3.1 Model facial information

In this stage a reference base is formed with information from individuals considered attractive. As the final result of the stage, a database of information of characteristics of attractive individuals is formed. The experiment used a list of 200 people known as the 100 most attractive women and the 100 most attractive men judged by the general public in 2014 by Maxim magazine [Max14]. The list was used for the construction of a base of three-dimensional points of interest meshes, where each model had its distances from its points extracted and stored for later use. Figure 5 present this stage overall.

Figure 5. Extracting facial information from a face mode

For each individual, a set of images are collected in different positions and facial expressions. The first iteration with the images is to keep the eyes in a horizontal alignment rearranging using transformation and scale matrix as Figure 6.

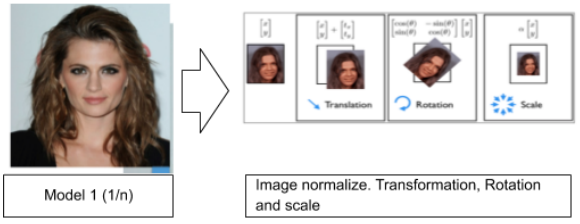


Figure 6. Image normalize process

In the second step, techniques of face Detection and extraction of points and distances are applied for each image to obtain characteristics of the individual. To extract the points of interest in two dimensions, the Face Tracker, developed by Sarigh, was used, based on the Haar-like pattern recognition process [Sar12], (see Figure 7). The Posit [Sar12] technique was used to estimate the points in three dimensions.

ISBN 978-80-86943-38-1

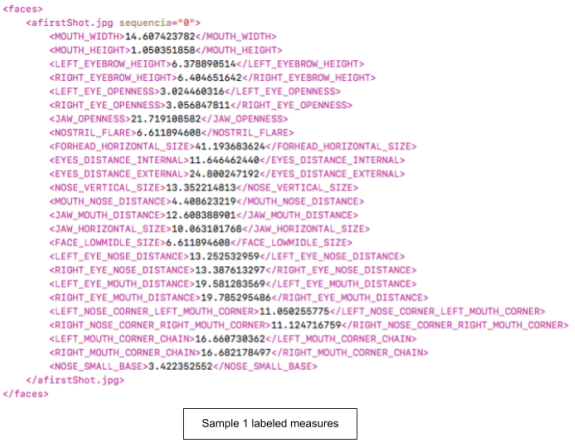


Figure 7. Labeled measures sample

The points of interest found, (see Figure 8) as example model 1 with maped face components and labeled, were triangulated according to the application of The Delaunay algorithm described by [Gui92] , thus constituting a mesh of points of interest.

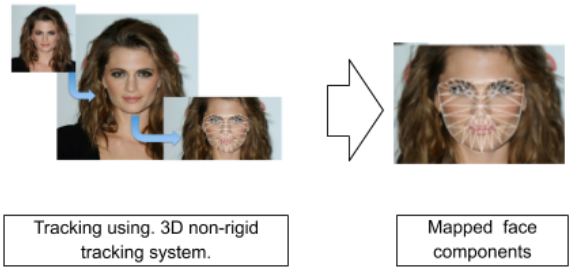


Figure 8. Mapped face components

Each extracted meshes dimensions are stored for use in the later stages.

3.2 Preparing and Prediction

The goal of this stage is to identify a similarity of any individual to some other of the reference basis of individuals considered attractive. In a video with some individual, sample images are extracted to identify the characteristics. For each image, techniques of face detection and extraction of points and distances are applied to obtain characteristics of the individual. The characteristics of the individual are compared with the characteristics of each individual which form the reference basis for information on characteristics of attractive individuals. The objective of the comparison is to identify the set of characteristics that most closely resembles the individual who wishes to apply the improvement of attractiveness. At the end of the stage a more similar and attractive face model is identified for the individual concerned.

In our implementation (Figure 9), the processing for comparison of information was done using KNN (K Nearest Neighbor) and SVM (Support Vector Machines) [Zha2006] data classifiers. The classifier's input for training is the file of meshes extracted from the model facial information stage.

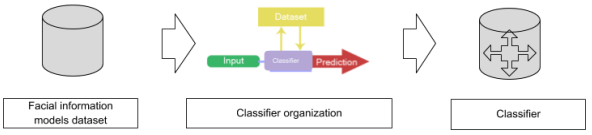


Figure 9. Classifier organization

The prediction (Figure 10) aims to answer which model most resembles a mesh of points of interest from any sample. For this purpose, the classifier has as input the points and distances of the face, and as output the indication of the individual that has the most similarity with the input face. The classifier has been configured in advance using the basis of attractive features.

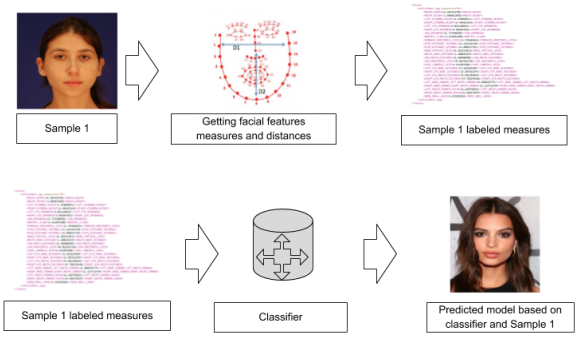


Figure 10. Prediction process

3.3. Transformation

Once the set of characteristics with greatest attractiveness and similarity with the face to be enhanced is identified, a geometric transformation is applied to all the video frames whose face is to be enhanced. As a result (see Figure 11 below), the individual has the same size and shape of components of the face (mouth, nose and eyes) according to the face model with more similar and attractive features. In our implementation, we applied the technique of warping points in image and blending textures, based on the work [Guo09],

ISBN 978-80-86943-38-1

[Lon17], [Con19] and [Cne17] that use so-called points of control to perform the distortion of the images, which were transformed to the points of the individual according to the model characteristics more similar and attractive identified.

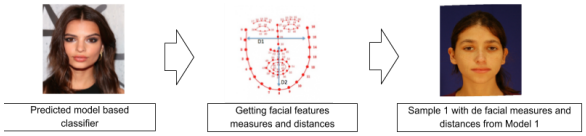


Figure 11. Geometric transformation process

For gaining scale in frame-by-frame processing of video and combine textures the process uses shape extraction based on the mesh of points of interest. The technique is based on the work of Saragih [Sar12] the process was evolved to perform the extraction of a the 3d mesh from a face identified in a video. Once the face is detected, it can be observed that its edges form a geometric shape around the mesh, it is made through the connection of the points and the texturing extracted from the frame in the delimited area. The geometric shape with the applied texture is called mesh (see Figure 12).

Figure 12. Mesh extract

4. EXPERIMENTAL RESULTS

Some results of our approach are presented in Table 1 which shows original videos in second column and processed videos in the third column.

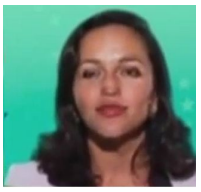
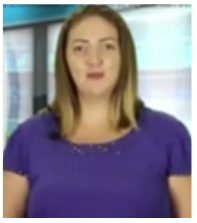
Model	Unmodified video	Modified video
Model 1	 https://youtu.be/H3OYtewsxoc	 https://youtu.be/dG9DvDjQAzI
Model 2	 https://youtu.be/h6aeP3WcW9E	 https://youtu.be/HWLe9_hqdQE
Model 3	 https://youtu.be/sHW49lc9-QU	 https://youtu.be/a-9upUQubQM
Model 4	 https://youtu.be/Ws15_nvSkGM	 https://youtu.be/izeIvMX7S6A
Model 5	 https://youtu.be/H7Fsuu_ExDs	 https://youtu.be/_mWUwarekvQ

Table 1 – unmodified x modified videos samples and links.

ISBN 978-80-86943-38-1

The videos was produced using a mac book pro late 2011, using 8gb of memory and a intel core i5 processor. The time of process to each frame 640x360 resolution took in average 3,1s.

An assessment was made to identify whether there was a gain in attractiveness in the processed videos.

The evaluation process was done through a question form and applied individually to a group of people in a working environment. The selection of people was random and no group received special attention. The applied form was online and the answers stored in a database. The questions contained in the form were divided into two parts, being the first to identify gender and age, and the second to collect attractiveness notes given individually for a set of demonstrated videos.

The environment used was a closed room to preserve silence and avoid any external distraction. The site was equipped with a projection screen and artificial light. In this environment the total of responses collected was 96.

The application of the form was conducted by a host who followed the same dynamic, addressing person to person, inviting to participate in the evaluation process. Each guest evaluator was directed by the host to the room prepared for the application of the form. The host was responsible for narrating each question and collecting the answer. Below are the first two questions related to the first part of the questionnaire:

-What is your age range?

-What is your gender?

The subsequent questions were directed by the presenter as follows: "watch the video and answer the question below".

A total of five videos were presented, and for each, the host asked: "As for the person seen in video 1, on a scale of 1 to 7 being 1 for an unattractive person and 7 for a very attractive person, classify:" the videos presented contained a shot of a random person expressing himself in speech to the camera. The total amount of videos used throughout the process were 10, 5 being processed by the attractiveness improvement technique and the other 5 not processed. The 10 videos were distributed in 2 forms, namely A and B:

of the 5 videos of form A, 3 were processed by the attractiveness improvement technique and the 2 others were not. Form B was assembled with the same videos; however, processing was applied to the 2 videos that did not receive processing on Form A.

The application forms to applicants were alternated. For half of the group of applicants form a was submitted and for the other half Form B. The answers were gathered in a single response base and the results presented in Table 2 and in the graphic of figure 13.

Model	**Sum of notes when modified video**	**Sum of notes when unmodified video**
Model 1	137	95
Model 2	140	125
Model 3	127	11
Model 4	115	110
Model 5	140	104

Table 2 – Sum of notes assigned modified videos x not modified

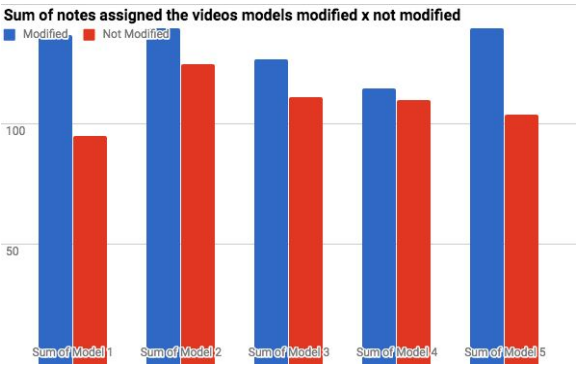


Figure 13. Graph result of the evaluation

5. CONCLUSION

The study presented demonstrates the use of computer vision and image transformation in video processing in favor of improving facial attractiveness. The automatic detection of points of interest along with the projection of three-dimensional forms provided the advance. Limitations found can be evolved using complementary technologies. The inaccuracy of the detection of facial points automatically can be combined with the use of three-dimensional cameras for better measurement avoiding flickers in de detection and inappropriate delimitation of features that cause a unexpected morphing transformation. Detection of mood and images of models in different expressions can also contribute to a better result.

ISBN 978-80-86943-38-1

Increasing the sample base of models with different face patterns can smooth out distortions given a face pattern incompatibility between base model and target model.

6. REFERENCES

[Let15] Leite, T. S.; De Martino, J. M. Improving the attractiveness of faces in images. Proceedings of the 28th Conference on Graphics, Patterns and Images - Sibgrapi 2015, Salvador, Bahia, Brazil,.

[Max14] Maxim Media. In: REVISTA. 2014. Available at: <http://www.maxim.com/women/2014-hot-100-post>. Access: 26 jun. 2018.

[Ley08] Leyand, Tommer et al. Data-driven enhancement of facial attractiveness.ACM Transactions on Graphics (TOG), v. 27, n. 3, p. 38, 2008.

[Eis06] Eisenthal, Yael; DROR, Gideon; RUPPIN, Eytan. Facial attractiveness: Beauty and the machine. Neural Computation, v. 18, n. 1, p. 119-142, 2006.

[Guo09] Guo, Dong; SIM, Terence. Digital face makeup by example. In: Computer Vision and Pattern Recognition, CVPR 2009. IEEE Conference on. IEEE, p. 73-79. 2009.

[Lon17] Lomas, Natasha et al. FaceApp uses neural networks for photorealistic selfie tweaks. 2017. Available at: <https://techcrunch.com/2017/02/08/faceapp-uses-neural-networks-for-photorealist

[Con19] Conger, Kate et al. The cost of hot selfie app Meitu? A healthy dose of your personal info. 2017. Available at: <https://techcrunch.com/2017/01/19/meitu-app-collects-personal-data/>. Access: 10 mar. 2019.

[Cne17] Constine, Josh In: Now Snapchat has “Filter Games”. 2017. Available at: <https://techcrunch.com/2016/12/23/snapchat-games/>. Access: 10 mar. 2019.

[Bran00] Bradski, Gary. "The opencv library." Dr Dobb's J. Software Tools 25 (2000): 120-125.

[Aba16] Abadi, Martín, et al. "Tensorflow: A system for large-scale machine learning." 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16). 2016.

[Sar12] Saragih, J., et al. Non-Rigid Face Tracking. Mastering OpenCV with Practical Computer Vision Projects, p. 189-233, 2012.

[Sla00] Slater, A., Bremner, G., Johnson, S. P., Sherwood, P., Hayes, R., & Brown, E. Newborn infants' preference for attractive faces: The role of internal and external facial features. Infancy, 1(2), 265-274, 2000.

[Lan87] Langlois, Judith H. et al. Infant preferences for attractive faces: Rudiments of a stereotype?. Developmental psychology, v. 23, n. 3, p. 363, 1987.

[Dia2019]Diamant, N., Zadok, D., Baskin, C., Schwartz, E., & Bronstein, A. M. Beholder-GAN: Generation and Beautification of Facial Images with Conditioning on Their Beauty Level. 2019.

[Gui92]Guibas, L. J., Knuth, D. E., & Sharir, M, Randomized incremental construction of Delaunay and Voronoi diagrams. Algorithmica, 7(1-6), p. 381-413, 1992.

[Fan2016] Fan, X., Chai, Z., Feng, Y., Wang, Y., Wang, S., & Luo, Z.. An efficient mesh-based face beautifier on mobile devices. Neurocomputing, 172, 134-142. 2016.

[Cho2012] Chou, J. K., Yang, C. K., & Gong, S. D.. Face-off: automatic alteration of facial features. Multimedia Tools and Applications, 56(3), 569-596. 2012.

[Dia2019] Diamant, N., Zadok, D., Baskin, C., Schwartz, E., & Bronstein, A. M.. Beholder-GAN: Generation and Beautification of Facial Images with Conditioning on Their Beauty Level. 2019.

[Zha2017]Zhang, L., Zhang, D., Sun, M. M., & Chen, F. M. (2017). Facial beauty analysis based on geometric feature: Toward attractiveness assessment application. Expert Systems with Applications, 82, 252-265. 2017.

[Zha2006] Zhang, H., Berg, A. C., Maire, M., & Malik, J. (2006). SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06) (Vol. 2, pp. 2126-2136). IEEE.2006

 ISBN 978-80-86943-38-1

ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

Immersive Analytics Sensemaking on Different Platforms

Sebastian Blum
Technische Universität Berlin
Berlin, Germany
s.blum@campus.tu-berlin.de

Gokhan Cetin
SIAT, Simon Fraser University
Vancouver, Canada
gcetin@sfu.ca

Wolfgang Stuerzlinger
SIAT, Simon Fraser University
Vancouver, Canada
w.s@sfu.ca

Abstract

In this work we investigated sensemaking activities on different immersive platforms. We observed user s during a classification task on a very large wall-display system (experiment I) and in a modern Virtual Reality headset (experiment II). In experiment II, we also evaluated a condition with a VR headset with an extended field of view, through a sparse peripheral display. We evaluated the results across the two studies by analyzing quantitative and qualitative data, such as task completion time, number of classifications, followed strategies, and shape of clusters. The results showed differences in user behaviors between the different immersive platforms, i.e., the very large display wall and the VR headset. Even though quantitative data showed no significant differences, qualitatively, users used additional strategies on the wall-display, which hints at a deeper level of sensemaking compared to a VR Headset. The qualitative and quantitative results of the comparison between VR Headsets do not indicate that users perform differently with a VR Headset with an extended field of view.

Keywords

Visual analytics, large high-resolution displays, sparse peripheral displays, virtual reality, head-mounted displays, spatialization, clustering, visualization, sensemaking, immersive analytics

1. Introduction

1.1. Visual Analytics

Visual analytics (VA) is a science-based activity supporting sensemaking for large, complex datasets through interactive visual data exploration [1]. In VA, users reason and make sense of the data through interaction with visualizations of the data. In the past, this concept gave rise to products such as Tableau, Microsoft Power BI, QlikView, and others.

In most VA applications, users interact indirectly with data through widgets, such as sliders and menus, which control the visualization through modifying the underlying model parameters. In contrast, semantic interaction enables analysts to spatially interact with their visualizations directly within the visual metaphor, using interactions that derive from their analytic process [2]. Further, semantic interaction supports sensemaking better than indirect interaction [3]. In a spatial workspace, users can directly arrange documents spatially into clusters to convey similarity or relationships in the data [4]. Spatial workspaces also allow users to establish implicit relationships in large datasets [5].

The space limitations of normal desktop displays lead to a need to remove/reduce/hide (at least temporarily) existing visualizations to make room for new ones, requiring an additional cognitive effort from the users to remember the now invisible information. By radically increasing display size, this dynamic could change substantially and allow users to visually access more information at once. Comparisons can then be made visually and directly rather than relying on memory and imperfect mental models – which supports the usability principle of recognition over recall. With desktop monitors, we often face a tradeoff between the level of detail and the number of different objects that can be displayed. New forms of displays, such as large display surfaces or virtual reality environments, can thus potentially improve the efficiency of VA activities. On a large display, a flick of the eye or turn of the head is all that is required to consult a different data source [6]. Previous work has found that large, high-resolution displays (LHRDs) improve productivity over traditional desktop monitors [6–8]. We can expect this to hold for VA applications on large displays as well. The exploration of different user interface technologies for data visualization applications has never been a core topic in information visualization [9].

The combination of Virtual Reality (VR) technologies, 3D user interfaces and VA systems is a new approach to analyzing large or complex datasets. This research field is nowadays described by the term Immersive Analytics [9,10]. Working with VR systems in a professional environment and using these for data visualization or as an immersive analytical workspace provides "*an easy and natural path to collaborative data visualization and exploration*" [11]] [p. 609] and shows possibilities to "*maximize intrinsic human pattern recognition*". Previous research has already shown that VR can support insight discovery in (primarily) spatial application domains and in helping to more effectively investigate brain tumors [12], MRI results [13], shape perception [14], underground cave structure analysis [15], geo-scientific [16,17], or

paleontology questions [18]. Thus, we expect LHRDs and VR systems to improve VA activities. The work reported here aims to compare these two approaches through user studies.

1.2. Sensemaking in Immersive Environments

How analysts make sense of a given data set is a crucial part of their work. The way humans understand and process information in VA activities is well described by the *Sensemaking Loop* [19]. This sensemaking loop breaks the process down into several stages, as illustrated in figure 1. For Immersive Analytics, i.e., doing VA activities in immersive environments, previous research shows how each stage of the sensemaking loop might be improved or impaired by the capabilities or limitations of the system [20].

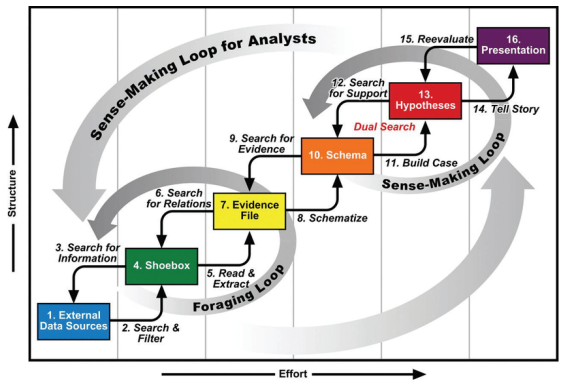


Figure 1: The sensemaking loop [19].

As seen in figure 1, the sensemaking loop involves a series of iterative steps for creating and evaluating a model for the data. Creating a model (bottom-up) involves finding information, extracting meaning, schematizing and building a case, and subsequently communicating that information. On the other hand, evaluating the model (top-down) involves re-evaluation, finding supporting evidence, finding relations in the information, or even finding basic information itself [19].

An integral part of the sensemaking process is foraging for information. To support such foraging, we could use represent data sources similarly to icons in an immersive environment, using either VR headsets or LHRDs. However in immersive environments, we can equip these representations with additional semantic meaning - such that the model of an engine might serve as a gateway for information about emissions, maintenance, power output, or other factors. One limitation of this approach is that there is much content that is not easily represented by icons, such as the cleaning budget for a department that sells cleaning products.

The second part of the sensemaking process is about synthesizing information, formulating hypotheses, and arranging supporting and contradictory evidence. Display space can play an important role in this process and assist in task completion, e.g., through the use of larger amounts of space for organization [6]. Further, the larger space offered by immersive systems provides a physical instantiation of the mnemonic device 'memory palace'. Thus, different parts of a complex model could be compartmentalized to different (virtual) spatial locations. Analysts can then use the space not only for their collected information but also to organize and structure their analytical workflow and thought processes [20]. In this paper we focus on this aspect of sensemaking.

Some VA tasks require specific types of interaction methods, e.g., the method to select data during foraging might be different from that to express a hypothesis. Good user interfaces for immersive environments are subject to specific guidelines and concepts. Elmqvist's fluid interaction concept distinguishes different interaction types for selection, filter, sort, navigation, reconfiguration and labeling and annotating in VA tasks [21]. For example, techniques such as mouse interaction are not appropriate in immersive environments where the user is standing and not sitting in front of a display. Gestural interaction, e.g., pointing to and circling one or more elements, could be used instead. For such interaction, designing comfortable gestures is necessary to minimize the strain on the user.

As discussed above, there is great potential for integrating the VA process into immersive environments. Thus, we need to examine the specific components of the sensemaking process that can be enhanced by immersive technologies and use the advantages of such technologies to support the VA sensemaking process. Here we focus on the ability of the user to arrange the visualizations of data on a large display canvas.

Grouping the information and generating clusters of related data is a part of sense-making in analytical tasks, which helps in the search for a way to encode data in a visual representation that helps to answer task-specific questions [22]. It takes place in the early, preparation stages of VA processes and its effectiveness can affect the efficiency of all following stages.

A core inspiration for this study was Endert et al.'s work [23], which presented the concept of semantic interaction that seeks to enable analysts to spatially interact within their analytical workspace. Following their work, we also believe it is important to observe how users organize their information, unaided by any algorithm, as this reveals insights about the human ability to understand large amounts of information through an interactive system. Thus, we do not use algorithmic clustering.

1.3. Large Display Systems

Large display systems are both qualitatively and quantitatively different from traditional displays. Reda et al. presented the results of a small-scale study to understand how display size and resolution affect insight. Although their results verify the generally accepted benefits of large displays, they also provide mixed results for extant work and propose explanations by considering the cognitive and interaction costs associated with visual exploration [24]. Other studies suggest that users working with large displays became less reliant on wayfinding aids in acquiring spatial knowledge. For example, Ni et al.'s experimental findings demonstrated the advantages of increased size and resolution [25]. As a general guideline, a LHRD was the preferred choice for IRVE applications,

 ISBN 978-80-86943-38-1

since it facilitates both spatial navigation and information gathering.

Our research uses multiple immersive display systems. One of them is a LHRD system called V4-SPACE. We used it to investigate how LHRDs can support VA tasks for a single user (figure 2). During this experiment we observed how physical size, resolution, and content spatialization on the workspace affect user performance on LHRD in VA tasks. A detailed description of V4-SPACE is given in the next chapter.

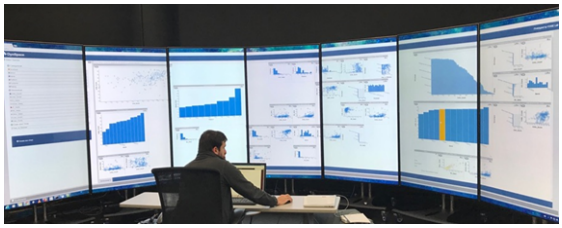


Figure 2: A user operating V4-SPACE

1.4. Head-Mounted Displays and Sparse Peripheral Displays

Extending VA system to be used within Virtual Reality (VR) systems poses challenges and opportunities. On the one hand, VR opens up new possibilities for data visualization, as virtual environments fully surround the user and the interface of the Human Machine Interface can be perceived to be more natural and intuitive. Thus, content can be placed and worked on all around the user. However, immersive VR technologies suffer from some restrictions, such as limited resolution, limited field-of-view, motion sickness, issues with interaction and gesture recognition, and with positioning objects in 3D space [9]. Even though current research in Immersive Analytics focuses primarily on low-level challenges [9], the above-mentioned obstacles still affect our research.

To address some of the above-mentioned issues of VR systems, we use a Sparse Peripheral Display HMD for our immersive VA study. The binocular field-of-view (FOV) of a standard VR HMD system is today about 84 degrees horizontal (110 degrees diagonal). By using a Sparse Peripheral Display (SPD), we can increase the field-of-view much by showing (low-resolution) content in the periphery of an HMD [26]. Increasing the FOV increases the spatial awareness of the user and might potentially even reduce simulator sickness.

A number of approaches and different systems to broaden the FOV through complex optical methods have been proposed. Fresnel lenses, such as the ones used in the StarVR and Wearality Sky, are challenging to manufacture and introduce optical distortions which are hard to mitigate [26]. Mirror based approaches [27] significantly increase the weight of their HMD devices.

In our study, we rely on previous research on SPDs [26] [28]. Xiao & Benko [26] presented a SPD that uses a lightweight, low-resolution and inexpensively produced array of LEDs, designed to surround the central high-resolution display of a Oculus Rift DK2 HMD. Their SPD expands the available FOV up to 190° horizontal. This SPD nearly fills the whole human FOV, which can span up to 210° giving the user a better perception of the virtual content that surrounds them (figure 3). Hashemian et al. [28] evaluated an improved SPD version in a HTC Vive setup, extending the Vives' FOV to 180° horizontally. They evaluated the performance of the SPD HMD in a spatial navigation study that used a previously presented navigational search paradigm [29,30]. Their results show that SPDs can provide a more natural experience of human spatial locomotion in virtual environments.

Figure 3: HTC Vive with a Sparse Peripheral Display extending the FOV to 180°.

2. A comparison of sense making behaviors in LHRD and HMDs

Based on knowledge about SPDs [26,28], large display systems [6,8] and immersive environments [9–11], combined with results from research about sensemaking [19,22] and productivity in VA [6,8,11,31,32], we performed two experiments in immersive analytics using different platforms. In Experiment I, we used a LHRD System and in Experiment II a SPD HMD with or without the SPD switched on.

To investigate sensemaking in each experiment and to compare how well both platforms used, we formulated the following hypotheses. We use the notation H1-II to signify the first hypothesis for the second experiment:

H1-I: Users spatialize their content through clustering information and exploit the advantages of a large display space.

H1-II: Users perform differently in a SPD HMD relative to an off-the-shelf VR HMD, resulting in different qualitative and quantitative outcomes.

H2-II: Users prefer a SPD over a standard VR HMD due to the larger FOV and thus a better sense of orientation in the space.

H3-II: SUS scores will show a higher sense of presence in the immersive environment with an SPD.

H1-C: Users perform better with a LHRD relative to a VR HMD, resulting in different qualitative and quantitative outcomes.

2.1. Global design decisions

To enable us to compare the results of both experiments, we aimed to maintain consistent design decisions across both experiments. Thus, we used (almost) identical experimental designs, using the same tasks for each experiment. We also tried to match the technological aspects of the HMD and the V4-Space as well as possible

 ISBN 978-80-86943-38-1

by setting up the curvature and size of the virtual screen displayed in the HMD to be similar to the one of the physical V4-Space LHRD.

2.1.1. Task

In both experiments users were given a workspace populated with charts. Users could create new charts on the workspace, delete existing ones, resize as needed, and move them around freely. In both experiments, the participants' objective was to solve a VA task in an immersive environment. Participants were asked to complete two tasks subsequently.

In the first task, participants were asked to arrange the scatterplots into groups based on *similarity*. They were told that they could consider any similarity criterion and group according to their own perception of the data. We enforced a minimum of at least five (5) groups, but they could create more. Participants were required to group all the charts. After completion of the grouping, they were asked to explain their motivation behind the way they grouped the charts. The responses were recorded about all aspects they considered.

In the second task, participants highlighted each dimension in turn through clicking on the data dimension in the left panel. When a data dimension on the data panel is clicked, each chart using this dimension in the visualization panel is highlighted. This feature let users see if their grouping criteria had a "common dimension", i.e. charts having the same dimension in one of their axes are grouped. The task was to observe any patterns that might become apparent.

We observed in pilots that the classification behavior of novice users was sometimes purely based on the chart type, which has no relation with the visualized data. To ensure that classification decisions were not affected by chart type, only a single data type (numerical) and a single chart type (scatterplots) were used in the studies. The dataset we used for experiment I and experiment II has been the same US Census dataset. However, to randomize the data over the conditions in the VR tasks, we used an additional second dataset in experiment II.

We also observed in pilots that VR controllers/wands did not provide sufficient performance. Thus, we decided to use the same input device, a high-resolution gaming mouse, in both experiments.

2.1.2. Experimental Design

We used a convergent parallel mixed methods design for both experiments to collect both quantitative and qualitative data. We acquired quantitative data by measuring the number of groups created in the clustering. Additionally, to gain qualitative data and to get insights into the participants' thoughts, we asked the participants to think aloud while they were performing the tasks.

In experiment I, each participant performed primarily the first task. We collected qualitative data to gain an insight into the participants behavior and of the potential advantages of a large display space.

In the second experiment, each participant was exposed to two conditions with the VR HMD: one with the SPD display on and another with the SPD switched off. This resulted in a within-subject design. The task described above was the same for each condition. Accounting for the repeated exposure to the task in the second condition and to consider its implicit learning, we randomized the order of the SPD/Non-SPD condition for each participant to minimize the bias and ensure counterbalancing. Further, we used two different datasets which we randomized over the SPD/Non-SPD conditions. Additional quantitative data for Experiment II were acquired by measuring the completion time of the first task.

Qualitative and quantitative data of both experiments were later used to investigate differences between the platforms. To avoid learning effects across both experiments, we used a between-subjects design to compare between the LHRD and SPD/Non-SPD HMD.

2.1.3.Data Collected

As part of the instruction phase for each experiment, participants filled in a questionnaire regarding their background and their knowledge of VA, VR and immersive environments like LHRD. Participants were asked to use the think-aloud protocol while performing the task and to share their thoughts about their clustering choices. Participants used the same protocol to share their observations about the highlighted dimensions in relation to their own grouping.

During the tasks, users' answers were recorded via free-form text boxes and were stored in a survey system. Users' clustering activity was tracked in two ways: First, we screen-recorded their activities. Second, we recorded the positions and information of each chart for each final participant's grouping.

To be able to better understand the raw screen recordings, we watched the user carefully at each step of the procedure and recorded observations, e.g., through jotting down signs of frustration at specific points in time, which further informs the analysis of the results.

At the end of the study, we performed an audio-recorded post-study interview with each participant, taking typically less than 5 minutes. This interview was semi-structured. We asked participants a list of questions regarding the tasks they completed, the software tool, the system, what they liked, what was challenging and/or confusing, and whether they had any further feedback.

2.2. Apparatus

The most notable components of the apparatus for this study are a LHRD for the first experiment and a HMD with/without SPD for the second experiment. We further used a VA tool called DynSpace to carry out analytical tasks. DynSpace runs as a web application on a Node.js server with MongoDB running in the backend to support data collection during experiments [33]. We chose a web application because of its ubiquitous accessibility and flexibility in terms of both client and server side technologies and to ease for future expandability on large display screens.

 ISBN 978-80-86943-38-1

2.2.1. V4-Space

V4-SPACE consists of a 1x7 array of large, tiled displays, each a vertically oriented 85" 4K Samsung Smart TV. The user sits in front of a desk at the center of the semi-circular arrangement of the TVs, see figure 2. We put an additional 21" monitor for auxiliary tasks before the user, below the line of sight to the large display. The main display has 15120 x 3840 pixels, which makes V4-SPACE a 58-megapixel system with 52 PPI pixel density. While the aspect ratio of a single display is 9:16, the system ratio is 63:16, about 4:1. The main display is 7.41 m by 1.88 m. In the 1x7 grid, there are no horizontal bezels and only 6 vertical ones.

V4-SPACE is controlled by a single computer with an Intel i7-6700K 4GHz processor with four PCI Express Gen 3 slots. The displays are driven by two nVidia Quadro M5000 cards, which provide four 4K outputs each, hardware synchronized through an nVidia Quadro Sync card. The auxiliary desktop monitor (which was not used for VA activities, and solely served to display a questionnaire form) is connected via an nVidia Quadro K620 graphic card. V4-SPACE relies on the nVidia Mosaic driver functionality, which presents all seven large displays as a single display to the user.

The system is designed for a single user who has a fixed position in front of the display system, about 3.3 m from each monitor. At this distance V4-SPACE is a "super-retina" display[34]. The display system is arranged in a circular arc (approximately 131° horizontal field-of-view, FOV) such that each monitor is equidistant to the user. This avoids information legibility issues due to non-uniform distances. As a limited form of physical navigation, the user can simply rotate their head or rotate a swivel chair to look at different parts of V4-SPACE.

Interaction is through keyboard and mouse. To support the high resolution of V4-SPACE, we use a Razer DeathAdder Chroma 10000 PPI optical gaming mouse, which permits the user to perform pixel-accurate pointing on the large display surface.

2.2.2. SPD HMD

For the HMD we used an off-the-shelf HTC Vive VR headset which was extended with a custom built Sparse Peripheral Display (SPD) to broaden the field-of-view of the virtual environment (figure 3). The SPD consists of 256 RGB LEDs placed on a 2-layer flexible printed circuit board. The LEDs are set in a radial array and the flexible PCS is folded to tightly fit around the lenses of the headset. The LEDs themselves are controlled by two LED drivers (Texas Instrument TLC5951), which provide 12-bit PWM color control to each color of the LED RGB triplet (36 bits of color per LED). Its LED drivers are controlled by an Cortex-M3 microcontroller (Cypress CY8C5288 LTI-LP090), which handles all communication with the computer through USB. The SPD update rate is set to 100Hz and the horizontal display FOV consists of approximately 180° (see [28] for a similar setup). We switched the SPD on and off for the two conditions investigated in this experiment.

2.2.3. Common software platform

The VA tool DynSpace is a browser-based VA tool written in JavaScript. Its user interface consists of two two panels: a Data Panel on the left side showing data dimensions and the main visualization panel on the right. Users can select data dimensions in the left one and drag-and-drop them into charts in the main visualization panel for analysis.

The visualization panel contains data charts that show relations between selected data dimensions. Each chart is contained in a rectangular sub-panel that a user can move, resize, add, or delete, and shows a 2D data plot. All plots are coordinated through brushing and linking.

DynSpace was designed to aid analysis of complex datasets [34]. Initially it displays a number of charts, each generated automatically by picking random pairs of data dimensions. This set of initial charts uses about half of the workspace in either of the experiments. The initial display of charts is a simple array with no clustering. Enough free space is left for the user to arrange the spatial layout of the content as they wish, e.g., by moving charts and creating clusters.

DynSpace uses a grid-based layout manager that enforces complete visibility of charts at all times by not allowing charts to partially or completely overlay one another. The available space is divided into invisible rows and columns so that the clusters always appear as an array.

3. Experiment I

The purpose of experiment I was to observe user behavior during a classification task in a LHRD.

3.1. Methods

The spatialization task used a simplified version of DynSpace to display 2D plots visualizing relations among a subset of the 2016 US Census Dataset. This study took place on the V4-SPACE display (figure 4).

Figure 4: The instance of DynSpace that spans the display space of our LHRD. Each square is a chart of a part of the data.

3.1.1. Participants

There were nine participants, P0 to P8, four of whom were male. Ages ranged from 18 to over 40. Some were undergraduate students participating for course credit, others were volunteers with at least a bachelor's degree. Through pilot studies, we ensured that all users could read the text on the displays from the default chair position without problems.

Participants were asked about their familiarity with seven VA terms and concepts in the pre-study survey. On average, they were familiar with 4.4 of those.

Since the study did not require specific domain knowledge, we were able to use mostly novice users for this study. 5 participants out of 9 did not have any VA experience. One reported less than a year of experience and three reported 1 to 3 years of experience. Seven of them had never used any

 ISBN 978-80-86943-38-1

visualization or VA tools, whereas one had used D3.JS and another used R in a statistics course for a term.
In a pre-study survey we asked users whether they could interpret data from scatter plots. They ranked their ability of interpreting data correctly from a scatter plot on a scale from 1 to 5. The results were 3x "Sometimes", 4x "Mostly" and 2x "Yes, always" answers; which respectively stood for 3, 4 and 5 on the scale used.

3.1.2. Procedure

We first instructed participants that they could rotate/swivel the chair to see the full display, but that the chair had to remain in the same location until the end of the experiment, to retain the same distance to all displays in the system. Leaning back and forward was permitted, as desired. Next, we trained them in basic, yet frequent operations on V4-Space: keeping track of the cursor, how to find it when they lost track of it, and switching between the LHRD and the auxiliary monitor. Then, users were introduced to DynSpace along with some practical tips regarding the usage. We assured users that they could ask questions during the experiment and that they should communicate their thoughts around the tasks or whenever they experienced issues. We recorded any direct or indirect feedback from the users during the tasks through note-taking.

3.2. Results

Reporting the results for experiment I, we use the notation P3-I to signify participant three from the first experiment.

3.2.1. Clustering

Three participants created only the minimum allowed five groups and four clustered their plots into 6 groups. The other two participants created 8 and 16 groups respectively.
The participants used one of the following three equally prevalent strategies when clustering: Similar visual appearance (P1-I, P2-I, P7-I), commonality in labels (P0, P4-I, P5-I), or common topics (P3-I, P6-I, P8-I). When building clusters, charts were either arranged horizontally or vertically (P0-I, P1-I, P4-I, P5-I), roughly circular (P6-I, P8-I), or in a mixed arrangement (P2-I, P3-I, P7-I).
For some users, there was no perceptible relation between different clusters in the workspace (P0-I, P1-I, P4-I, P5-I). For P3-I, cluster shapes were determined by cluster types. For the rest (P2-I, P6-I, P7-I, P8-I), the distance between clusters decreased as the similarity between clusters increased. For these participants, relations between clusters were reflected by cluster separation. For some participants, there was either no (P0-I, P1-I) or only a minimal (P3-I, P5-I, P8-I) distance between clusters. For some participants (P4-I, P6-I) the bezels strongly influenced the cluster arrangement. For a few users (P2-I, P7-I), the distance between clusters varied depending on the inter-cluster relations.

3.2.2. Space Usage

Out of nine participants, seven used the entire width display provided by the system. If a user runs out of free space, DynSpace permits vertical scrolling via the mouse wheel (but not horizontal scrolling). Surprisingly, P0-I and P1-I used only about 2/7 and 3/7 of the space, respectively, and relied heavily on vertical scrolling.

3.2.3. Navigation Techniques

Six of the participants used physical navigation frequently during the tasks, i.e., they rotated their head and/or body back and forth to "access" all parts of the LHRD visually. The others kept their gaze mostly focused on a subset of the displays.

4. Experiment II

The purpose of Experiment II was to observe sensemaking in a VR environment and investigate whether an increased field of view in VR via a SPD has an effect on VA tasks.

4.1. Methods

We used a HTC Vive with a SPD to show the virtual environment through Unity 3D. The VR environment consisted of a curved display surface, as seen in figure 5.

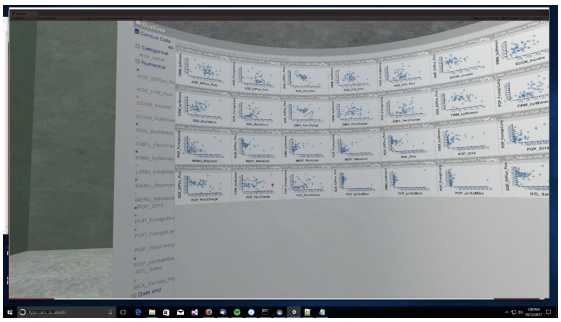


Figure 5: Curved screen surface in the virtual environment showing the scatterplots.

We duplicated several desktop windows into the virtual environment and displayed different web applications, including DynSpace, using the Awesomeium plugin for Unity 5. We also redirected mouse and keyboard events to enable interaction with the content of these windows. We displayed more than a single window to make sure that participants could not get lost in the virtual environment. The largest window was placed in front of the user and showed DynSpace with 64 scatter plots on the VR surface to perform the VA task. Participants used a computer mouse to interact with the DynSpace user interface which was placed on a table in front of the participants. We used a screen-capture software to record the participants' interactions with the system and to collect the task data.

4.1.1. Procedure

Prior to the study, participants were instructed in the specifics of VR, SPD and VA systems and were asked to give informed consent. They were further asked to agree to screen capturing and audio recording while taking part in the study. Participants were verbally instructed on the procedure for the experiment and exposed to a short training session prior to the actual task. A researcher was always present to collect observations, provide instruction, and for technical support, if needed.

 ISBN 978-80-86943-38-1

Participants sat on a chair located in the middle of a lab environment, with a little table in front of them to use the computer mouse. The surrounding area was emptied so they could not touch or bump into anything while being immersed to the virtual world.

After each condition, each participant completed a Slater-Usoh-Steed questionnaire to account for their perception of presence. This has been additional collected data compared to experiment I.

4.1.2. Participants

In total, 7 (new) subjects (5 female; mean age was 21.28 ± 2.05 years) participated. Subjects were students of the local university and recruited from a local subject pool. The participants were unaware of the purpose of the study. The experiment was approved by the ethics committee of the university.

A pre-study survey indicated that the participants had little experience in VA tasks (3x "None", 3x "less than a year", 1x "1-3 years") and have been uniformly familiar with VR Headsets (3x "Yes", 4x "No").

4.2. Results

For the results of experiment II, we use the notation P4-II to signify participant four from the second experiment.

4.2.1. Completion Time

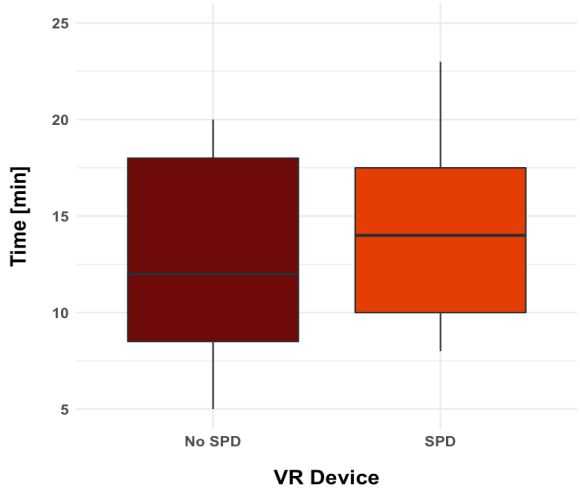


Figure 6: Boxplots showing the time needed to complete the subtasks in either the SPD or the no-SPD condition.

No participant exceeded the 60 minute limit of the study. Completion time of each task varied from 5 to 20 minutes in the No-SPD condition (M = 12.85, SD = 6.12) and from 8 to 23 minutes (M = 14.28, SD = 5.40) in the SPD-condition (see figure 6).

A Shapiro-Wilk test showed a normal distribution for both, the No-SPD and the SPD condition. Variance homogeneity between conditions exists. An ANOVA showed no significant results of the condition on the working time ($F(1,6)$ = 3.614, p = 0.105, η^2 = 0.018).

Participant P7-II created by far the biggest number of groups (SPD: 15, No SPD: 17). The participant noted afterwards, that they wanted to solve the task as soon as possible. Since the participant only relied on the commonality of labels, they performed merely a pattern recognition task on the naming of the plots, rather than truly making sense of the data.

In contrast, participant P3-II solved the task very carefully and even changed their grouping strategy in the second condition. They took about 20 minutes for grouping through "commonality in labels" (No SPD) and 23 min. for grouping by similar visual appearance (SPD).

4.2.2. Number of Clusters

The number of groups (figure 7) created in each condition varied from 6 to 17 in the No-SPD condition (M = 9.42, SD = 3.69) and 6 to 15 in the SPD condition (M = 8.42, SD = 3.20). A Shapiro-Wilk test showed a normal distribution for the No-SPD condition and not for the SPD condition. Variance homogeneity between conditions exists. We performed a parametric test since it is adequate robust to possible violations of the normality assumption [35]. The ANOVA identifies a significant difference between conditions for the number of groups created ($F(1,6)$ = 7.000, p = 0.0382, η^2 = 0.024).

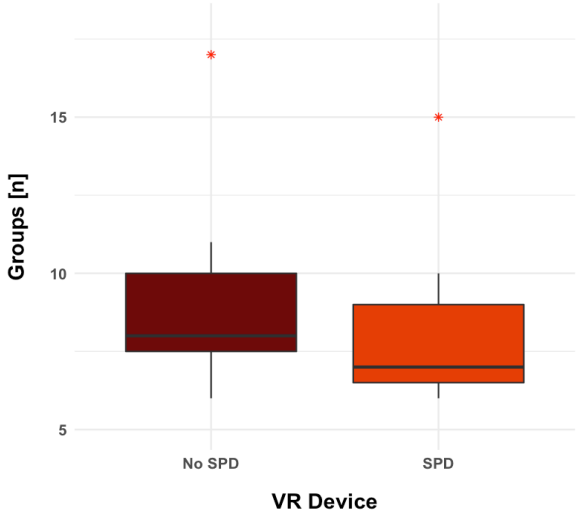


Figure 7: Boxplots showing the number of clusters in each subtask in either the SPD or the no-SPD condition.

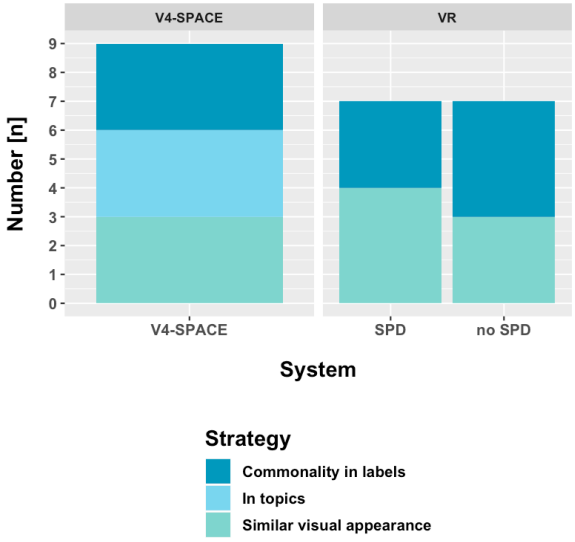
4.2.3. Clustering Strategy

Participants used two different strategies for grouping plots in experiment II. The first strategy was to group by a commonality in labels. The second one was to group by a similar visual appearance of the scatterplots shown, e.g., (broadly) increasing plots vs. plots with no visible trends. While we expected that participants would use the latter strategy more frequently, the first one was used more widely. We could not identify a considerable difference of the clustering strategy between the SPD and No-SPD conditions.

However, we observed a shift in strategy, from an initial grouping by commonality in labels to a grouping by similar visual appearance (illustrated in figure 8). This shift seems to be because of the succeeding subtasks. After the first subtask, one participant (P3-II) mentioned that they expected the visual appearance of the scatterplots to be more important for making sense of the data, rather than just ordering them by a commonality in axis labels. Thus,

ISBN 978-80-86943-38-1

this participant grouped the graphs in the second condition according to the visual appearance of the scatterplots.Figure



8: Barcharts showing the strategy used for clustering in V4-SPACE and VR setups (for both experiments).

4.2.4. Cluster Shapes

We observed three different kinds of arrangements participants used to classify the data: horizontally and vertically arranged, roughly circular, or a mixed arrangement (see figure 9). This classification is similar to arrangements observed in experiment I

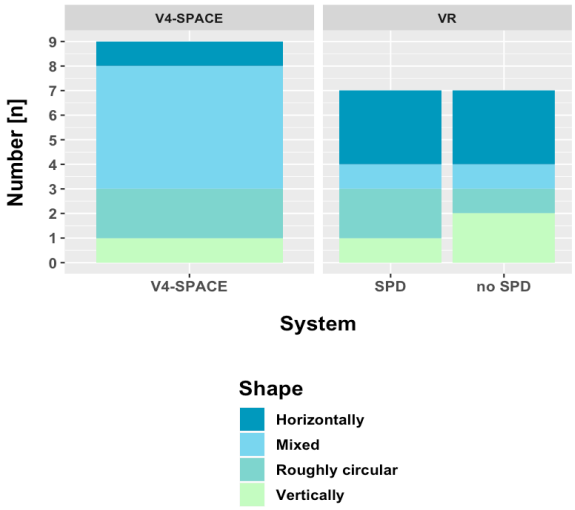


Figure 9: Barcharts showing the shape of the clusters used for clustering in V4-SPACE and VR setup

Interestingly, the arrangement of clusters seems to directly correspond to the used clustering strategy. When grouping by commonality in labels, the arrangement of the clusters was horizontal or vertical. The strategy to use similar visual appearances changed the arrangement of the cluster shapes to a more "prototypical" arrangement of clusters with roughly circular shapes. Participant (P3-II) even commented that this seemed to be „more intuitive". This indicates that the arrangement of clusters may be directly related to the clustering strategy

4.2.5. Advantages and Disadvantages of Sparse Peripheral Displays

Asking the Participants what HMD condition they preferred, the outcomes reveal the same equal level of preference for "No SPD preferred", "No difference experienced" or "SPD preferred" (see figure 10). One participant (P1-II) noted that they did not notice the SPD change between the tasks.

Those participants who did not like the SPD condition mentioned that they perceived the SPD display to be too bright and thus as distracting while solving the VA task. One participant (P7-II) felt that the SPD impaired perception and that it actually "took away" from the experience. The same participant mentioned that the SPD was "some sort of separately perceived reality" to them.

When asked about the SPD experience without a focus on solving a VA task, the SPD was preferred by a majority of participants, potentially due to the perception of less or no motion sickness. This result is contrary to the results of [28], where they observed an increase in motion sickness in using a SPD. Further, one participant (P3-II) mentioned that the SPD gave him a better sense of orientation. This helped him to navigate better between the content in the second subtask (the highlighting of the dimensions).

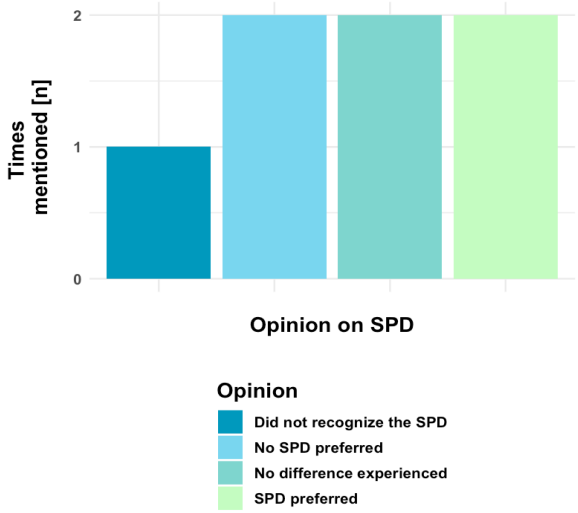


Figure 10: Preferences of the participants regarding the SPD headset.

4.2.6. Presence Questionnaire

The overall Slater-Usoh-Steed (SUS) score for each subject was quite high, resulting in a high prevalence of participants being present in the virtual environment in general (figure 11). An ANOVA identified no significant results of the condition on the SUS score ($F(1,12) = 0.025$, $p = 0.878$, $\eta^2 = 0.0021$). Thus, our data disconfirm hypothesis H3-II.

Only a single participant (P7-II) gave the SPD condition a conspicuously higher SUS score than the no-SPD condition. The participant is the same who identified the SPD to be some sort of separately perceived reality (see section 4.2.5). Interestingly, the same participant also mentioned the SPD to be less preferable than the No-SPD condition, which contradicts their SUS rating.

ISBN 978-80-86943-38-1

P3-II identified the SPD condition as more preferable and that it gave them a broader sense of orientation during the task. Looking at the participants' SUS scores, there was no significant difference between the conditions though. Overall, the SUS scores of P3-II were notably lower than any other participants' scores.

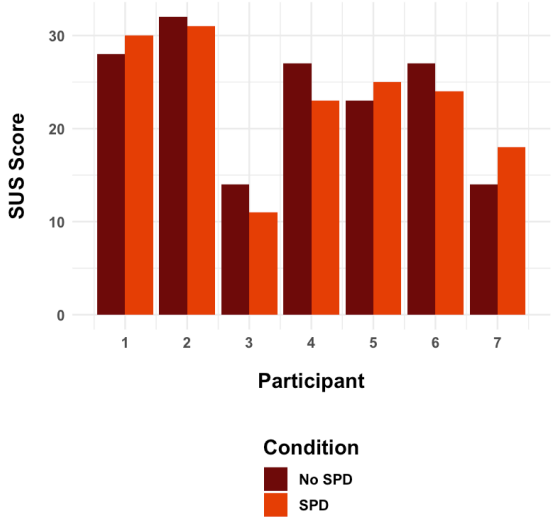


Figure 11: Results of the Slater-Usoh-Steed questionnaire for each participant and each HMD condition.

4.2.7. Highlighting of Dimensions

The highlighting of the dimensions was a solely qualitative task. For those participants who grouped the plots by a commonality in labels, any highlighting of the dimensions directly matched the arranged groups. It is interesting to see that even though the participants did not know the requirements of the second subtask at the beginning, they arranged the groups by the dimensions in the first subtask. This was helpful for the (future) second subtask. Yet, no participant discovered and mentioned this. Most of the participants did not identify any reasons as to why some dimensions matched their groupings and why some did not. Not surprisingly, the participants who grouped according to similar visual appearance tried to make sense of their arrangement far more often than the participants who grouped by the commonality in labels. Their groups were arranged more likely by different dimensions and matched the highlighting less well. Participant (P5-II) mentioned it might have been better for the second subtask to arrange the graphs by the commonality in labels as opposed to arranging the clusters in a roughly circular manner. Using the think-aloud protocol, most participants who grouped by similar visual appearance mentioned the potential utility of cross references between the arranged groups according to the dimensions. Participant (P6-II) arranged the graphs by commonality in labels. However, faced with the second subtask, the participant tried to explain the dimensional matching only based on the scatterplot appearance and did not rely on the labels. However, the participant was still not able to come up with a reason as to why some plots where in different groups in their arrangement.
Overall, most of the participants found the second subtask of making sense of the data and highlighting the dimensions challenging. One potential explanation is the relative lack of experience with VA in our participants.

5. Discussion

We can confirm Hypothesis H1-I, that users spatialize their content through clustering information and exploit the advantages of a large display space, through the observations of the users' clustering behaviour in experiment I. Instead of treating the plots as a single group of data, they classified the plots through various strategies and used the full display space during sensemaking to create different clusters of data plots.
Our second experiment, does not support any of the hypotheses H1-II, H2-II or H3-II, as we did not find any significant results that users perform differently in a SPD HMD compared to a common VR Headset.
Combining the quantitative results for completion time and number of clusters with the qualitative data of clustering strategy and cluster shapes, we are unable to discern a clear picture if users perform differently in a SPD HMD relative to an off-the-shelf VR HMD. Thus, our hypothesis H1-II is not supported.
We see no support for hypothesis H2-II, since we are unable to reach a clear conclusion on whether the SPD condition was preferred by the users while solving a VA task. The (low-resolution) larger field-of-view did not seem to have had a strong effect on the outcome. Finally, we observed no difference on the Slater-Usoh-Steed questionnaire, resulting in an unmet hypothesis H3-II. Using a between-subjects design we compared the results of experiment I and experiment II to investigate differences in sensemaking between the LHRD system and the VR HMD with and without the SPD. As seen in figure 8, 12 and 9, there are several differences between the results of experiment I and experiment II.

As we did not record the completion time in the first Experiment, we were not able to compare times between the experiments. We thus analyzed quantitative data in terms of the number of created clusters to compare both experiments. A Shapiro-Wilk test showed a normal distribution for the No-SPD condition, but not for the SPD and the LHRD. Variance homogeneity between the three exists. Since parametric tests are robust to the violation of the normality assumption [35], we performed an ANOVA over V4-SPACE and HMD results (V4-Space: $M = 7.0$, $SD = 3.5$; No-SPD condition: $M = 9.42$, $SD = 3.69$; SPD condition: $M = 8.42$, $SD = 3.20$). The results showed no significant difference between the number of created cluster in V4-Space and the no-SPD HMD ($F(1,12) = 1.257$, $p = 0.284$, $\eta^2 = 0.095$) and the V4-SPACE and the SPD HMD ($F(1,12) = 0.449$, $p = 0.515$ $\eta^2 = 0.036$).
Thus, we may disconfirm hypothesis H1-C and can only state that users do not seem to perform differently in an HMD relative to a LHRD. Users did not perform better in a LHRD relative to a VR HMD according to our investigations.

ISBN 978-80-86943-38-1

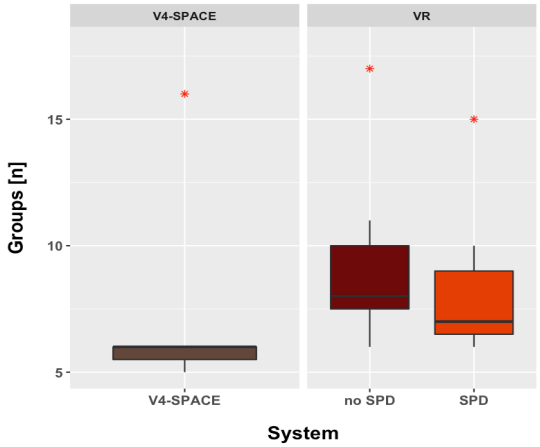


Figure 12: Number of clusters generated in V4-SPACE and the VR setups.

Clustering Strategy

In terms of the clustering strategy, in the second experiment a subset of users relied on grouping based on the commonality of labels. Yet, as the participants progressed through the experiment, we observed a transition towards a strategy based on similar visual appearance of scatterplots in some participants. This could be considered to be a more reflective way of grouping the plots. We hypothesize that grouping by similar visual appearance corresponds to a higher-level sensemaking activity, since it involves more active reflection about the data. In the first experiment, we observed an additional, third strategy. While about a third of the participants based their strategy on the visual appearance of the scatterplots and another third based it on the commonality of axis labels, the remaining third adopted a clustering approach based on common topics of plots. This strategy is more complex and reveals that some were actively trying to make sense of the data.

Using this strategy, participants neither clustered according to a pattern recognition/visual matching strategy, regardless of what the data presents; instead, they aimed to bring charts together based on what the data might be about. Even if the axis labels were different, users thought about the concepts and what those concepts are about and clustered the plots so that each went into a corresponding category of topics.

Cluster Shape

We also compared the shape of the clusters the participants generated. In the first experiment the participants' rearrangement efforts resulted mostly in simple shapes, either highly rectangular (aspect ratio substantially different from 1) or roughly circular (aspect ratio around 1). Classifying cluster shapes by aspect ratio (large ratio: horizontal; small: vertical) yields the following grouping (with examples shown in figure 13):

- Horizontal: 1 participant.
- Vertical: 3 participants.
- Roughly circular: 5 participants.

For several participants, no clear pattern of cluster arrangement could be identified. P0-I and P1-I did not have any space between clusters arranged in a random order. For P4-I and P5-I, clusters were separated by some distance. However, the spacing seemed uniform and clusters appeared similar. In contrast, other participants used cluster position to convey meaning. As an example, P3-I used the cluster shape to indicate what the clusters represented. While P3-I categorized groups looked somewhat similar, their group of uncategorized plots looked different from other clusters in terms of shape and the distance from other clusters.

We observed similarities in cluster shapes in experiment II as well. Participants' cluster shapes were defined similarly to the first experiment (horizontal or vertical, roughly circular, mixed arrangement). As in experiment I, horizontal or vertical arrangement was used by participants that did not seem to think deeply about their clustering strategy.

When users appeared to make more sense of the data, they started to create more traditional "roughly circular" clusters and used shape and space as a tool for sensemaking within the available workspace.

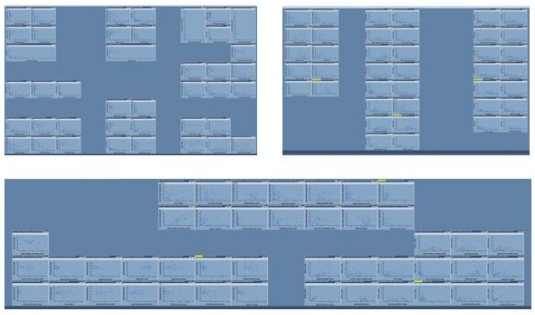


Figure 13: Variation in aspect ratios used by participants. P6-I used a roughly circular arrangement (aspect ratio ~1, top left) while P4-I used a vertically-oriented layout (aspect ratio <<1, top right) and P5-I a more horizontal organization (aspect ratio >> 1, bottom).

5.1. Summary

We conducted two sensemaking studies in immersive environments, one on a large high-resolution display (LHRD) called V4-SPACE and one using a HMD-based VR system with a sparse peripheral display option that we turned on and off. Using equivalent tasks, we observed users' behaviors in those environments and compared the outcomes within the same environment and the two separate display environments to each other.

The first exploratory study aimed to identify challenges and yielded qualitative observations for VA tasks in a LHRD environment. We obtained new empirical information about users' approaches when asked to prepare a large volume of data for analytical tasks on large displays. Even though all users started with the same state, the results of experiment I suggest that different users follow varying classification and spatial organization strategies. We observed clear distinctions in terms of clustering strategies, space usage, and preferred navigation techniques.

In the second experiment, we investigated the utility of a sparse peripheral display condition on a VR HMD. The presence of the SPD yielded mixed responses. While some users found it 'too bright', 'distractive' or thought that it reduces the immersion; others thought it causes less or no

ISBN 978-80-86943-38-1

motion sickness or that it supported a better sense of orientation. Quantitative analysis yielded a significant larger number of clusters created in the SPD than in a common VR HMD.

The comparison of both experiments showed no significant difference of the performed quantitative data in V4-SPACE and the VR SPD HMD. A comparison of the qualitative data yielded an additional strategy used in V4-SPACE and a slightly varying distribution of cluster shapes. The results show initial tendency that users perform different in both immersive environments, leaving the possibility for future research.

5.2. Limitations

A limitation we have to consider is the small sample size of n = 9 in experiment I and n = 7 of experiment II. While this limits the strengths of the insights we can derive, we point out that the effort to run the studies was high (and the SPD broke during the second experiment for the eight participant).Further, our participants were university students with limited VA experience. We observed that some participants in experiment II could not tell what was happening when highlighting dimension. We believe the lack of VA experience might have impacted how much sense the participants were able to extract from the data.

We tried to carefully match experimental conditions for both environments through using the same input device, apparent size of the workspace, amount of information displayed and the tasks that participants had to perform. Yet, there were also technological limits to this, such as the maximum resolution of the available VR headset system being far inferior to what is available on a LHRD.

Since the participants could terminate the task whenever they wanted, the duration to solve the task mostly depended on their motivation and their clustering and sensemaking strategies for the data. Further, since there was no single correct solution for the task, each participant used a different classification strategy with their own criteria. This limitation introduces noise into the main goal of comparing sensemaking in HMD and LHRD technologies and makes it more difficult to draw strong conclusions. In future work, it would be interesting to repeat the experiment on a conventional computer environment as a reference baseline.

A limitation for the second experiment in specific is using a computer mouse as an input device for the VR HMD. Yet, as mentioned above, we found that a VR controller was not sufficiently accurate to enable users to interact with the details of the charts. Since a mouse is not a typical HMD input device, it might make its interaction with the system unnatural. While participants could not see the physical mouse they were using while being immersed in the HMD, none of the participants reported any issues with this.

References

1. Shrinivasan YB, van Wijk JJ. Supporting the analytical reasoning process in information visualization. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM; 2008. pp. 1237–1246.

2. Endert, Alex, Patrick Fiaux, and Chris North. Unifying the sensemaking loop with semantic interaction. IEEE Workshop on Interactive Visual Text Analytics for Decision Making at VisWeek. 2011;

3. Endert A, Fiaux P, North C. Semantic Interaction for Sensemaking: Inferring Analytical Reasoning for Model Steering. IEEE Trans Vis Comput Graph. 2012;18: 2879–2888.

4. Endert A, Fox S, Maiti D, Leman S, North C. The Semantics of Clustering: Analysis of User-generated Spatializations of Text Documents. Proceedings of the International Working Conference on Advanced Visual Interfaces. New York, NY, USA: ACM; 2012. pp. 555–562.

5. Marshall CC, Rogers RA. Two Years Before the Mist: Experiences with Aquanet. Proceedings of the ACM Conference on Hypertext. New York, NY, USA: ACM; 1992. pp. 53–62.

6. Andrews C, Endert A, North C. Space to think: Large, High-Resolution Displays for Sensemaking. Proceedings of the 28th international conference on Human factors in computing systems - CHI '10. 2010; 55–64.

7. Baudisch P, Good N, Bellotti V, Schraedley P. Keeping Things in Context: A Comparative Evaluation of Focus Plus Context Screens, Overviews, and Zooming. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. New York, NY, USA: ACM; 2002. pp. 259–266.

8. Czerwinski M, Smith G, Regan T, Meyers B, Robertson G, Starkweather G. Toward Characterizing the Productivity Benefits of Very Large Displays. Interaction. 2003;3: 9–16.

9. Chandler T, Cordeil M, Czauderna T, Dwyer T, Glowacki J, Goncu C, et al. Immersive Analytics. 2015 Big Data Visual Analytics, BDVA 2015. 2015. doi:10.1109/BDVA.2015.7314296

10. Marriott K, Schreiber F, Dwyer T, Klein K, Riche NH, Itoh T, et al. Immersive Analytics. Springer; 2018.

11. Donalek C, Djorgovski SG, Cioc A, Wang A, Zhang J, Lawler E, et al. Immersive and Collaborative Data Visualization Using Virtual Reality Platforms. 2014 IEEE International Conference on Big Data Immersive. 2014; 609–614.

12. Zhang S, Keefe DF, Dasilva M, Laidlaw DH, Greenberg BD. An Immersive Virtual Environment for DT-MRI Volume Visualization Applications : a Case Study. IEEE. 2001;Vi: 1–5.

13. Chen J, Cai H, Auchus AP, Laidlaw DH. Effects of Stereo and Screen Size on the Legibility of Three-Dimensional Streamtube Visualization. IEEE Trans Vis Comput Graph. 2012;18: 2130–2139.

14. Demiralp C, Jackson CD, Karelitz DB, Zhang S, Laidlaw DH. CAVE and Fishtank virtual-reality displays: a qualitative and quantitative comparison. IEEE Trans Vis Comput Graph. 2006;12: 323–330.

15. Ragan ED, Kopper R, Schuchardt P, Bowman DA. Studying the effects of stereo, head tracking, and field of regard on a small-scale spatial judgment task. IEEE Trans Vis Comput Graph. 2013;19: 886–896.

16. Helbig C, Bauer HS, Rink K, Wulfmeyer V, Frank M, Kolditz O. Concept and workflow for 3D visualization of atmospheric data in a virtual reality environment for analytical approaches. Environ Earth Sci. 2014;72: 3767–3780.

17. Hsieh T-J, Chang Y-L, Huang B. Visual analytics of terrestrial lidar data for cliff erosion assessment on large displays. Satellite Data Compression, Communications, and Processing VII. International Society for Optics and Photonics; 2011. p. 81570D.

 ISBN 978-80-86943-38-1

18. Laha B, Bowman DA, Socha JJ. Effects of VR System Fidelity on Analyzing Isosurface Visualization of Volume Datasets. 2014;20: 513–522.

19. Pirolli P, Card S. The Sensemaking Process and Leverage Points for Analyst Technology as Identified Through Cognitive Task Analysis. Proceedings of international conference on intelligence analysis. 5: 2–4.

20. Stuerzlinger W, Dwyer T, Drucker S, Görg C, North C, Scheuermann G. Immersive Human-Centered Computational Analytics. In: Marriott K, Schreiber F, Dwyer T, Klein K, Riche NH, Itoh T, et al., editors. Immersive Analytics. Cham: Springer International Publishing; 2018. pp. 139–163.

21. Elmqvist N, Moere AV, Jetter H-C, Cernea D, Reiterer H, Jankun-Kelly TJ. Fluid interaction for information visualization. Inf Vis. SAGE Publications; 2011;10: 327–340.

22. Russell DM, Stefik MJ, Pirolli P, Card SK. The Cost Structure of Sensemaking. Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems. New York, NY, USA: ACM; 1993. pp. 269–276.

23. Endert A, Fiaux P, North C. Semantic Interaction for Visual Text Analytics. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. New York, NY, USA: ACM; 2012. pp. 473–482.

24. Reda K, Johnson AE, Papka ME, Leigh J. Effects of Display Size and Resolution on User Behavior and Insight Acquisition in Visual Exploration. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. New York, NY, USA: ACM; 2015. pp. 2759–2768.

25. Ni T, Bowman DA, Chen J. Increased Display Size and Resolution Improve Task Performance in Information-Rich Virtual Environments. Proceedings of Graphics Interface 2006. Toronto, Ont., Canada, Canada: Canadian Information Processing Society; 2006. pp. 139–146.

26. Xiao R, Benko H. Augmenting the Field-of-View of Head-Mounted Displays with Sparse Peripheral Displays. Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16. 2016. pp. 1221–1232.

27. Nagahara H, Yagi Y, Yachida M. Super wide viewer using catadioptrical optics. Proceedings of the ACM symposium on Virtual reality software and technology. ACM; 2003. pp. 169–175.

28. Hashemian AM, Kitson A, Ngyuyen-Vo T, Benko H, Stuerzlinger W, Member, et al. Evaluating a New Sparse Peripheral Display in a Head-Mounted Display for Vection and a Navigational Search Task.

29. Nguyen-Vo T, Riecke BE, Stuerzlinger W. Moving in a box: Improving spatial orientation in virtual reality using simulated reference frames. 2017 IEEE Symposium on 3D User Interfaces (3DUI). 2017. pp. 207–208.

30. Riecke BE, Bodenheimer B, McNamara TP, Williams B, Peng P, Feuereissen D. Do we need to walk for effective virtual reality navigation? physical rotations alone may suffice. Proceedings of the 7th international conference on Spatial cognition. Springer-Verlag; 2010. pp. 234–247.

31. Thomas JJ, Cook KA, Process AD. Illuminating the path: The research and development agenda for visual analytics. IEEE Computer Society. 2005; 184.

32. Bi X, Balakrishnan R. Comparing Usage of a Large High-Resolution Display to Single or Dual Desktop Displays for Daily Work. CHI '09 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. 2009. pp. 1005–1014.

33. El Meseery M, Wu Y, Stuerzlinger W. Multiple Workspaces in Visual Analytics. 2018 International Symposium on Big Data Visual and Immersive Analytics (BDVA). 2018. pp. 1–12.

34. Cetin G, Stuerzlinger W, Dill J. Visual Analytics on Large Displays: Exploring User Spatialization and How Size and Resolution Affect Task Performance. 2018 International Symposium on Big Data Visual and Immersive Analytics (BDVA). 2018. pp. 1–10.

35. Rasch D, Guiard V. The robustness of parametric statistical methods. Psychol Sci Q. 2004;46: 175–208.

 ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

Techniques for GPU-based Color Quantization

Matthias Trapp
Hasso Plattner Institute,
Digital Engineering Faculty,
University of Potsdam, Germany
matthias.trapp@hpi.de

Sebastian Pasewaldt
Digital Masterpieces GmbH
Germany
sebastian.pasewaldt@digitalmasterpieces.com

Jürgen Döllner
Hasso Plattner Institute,
Digital Engineering Faculty,
University of Potsdam, Germany
juergen.doellner@hpi.de

ABSTRACT

This paper presents a GPU-based approach to color quantization by mapping of arbitrary color palettes to input images using Look-Up Tables (LUTs). For it, different types of LUTs, their GPU-based generation, representation, and respective mapping implementations are described and their run-time performance is evaluated and compared.

Keywords: color palettes, color quantization, look-up tables

1 INTRODUCTION

1.1 Motivation and Applications

In computer graphics, a color palette P denotes a finite list of colors $P = C_0, \ldots, C_n$, each associated with a color space $C_i \in XYZ \subset \mathbb{R}^3$, with n being the total number of colors within a palette. In hardware or software palettes, n is limited by color bit-depth or number of simultaneously available total colors.

There are multiple applications for a fast color palette mapping approach supported by a Graphics Processing Unit (GPU)-based implementation. Besides enabling non-uniform color reduction, it can be used in stylized rendering or image and video "retro" abstraction operations that mimicking old devices, such as consoles.

1.2 Problem Statement

Given an *indexed image* $I[0,w] \times [0,h] \rightarrow i = 0, \ldots, n$ and a color palette P, a respective color mapping of an output raster image $G[0,w] \times [0,h]$ can be formulated as follows:

$$G[x,y] = \rho(I[x,y],P) \qquad \rho(i,P) = C_i \qquad (1)$$

where the *color mapping* function ρ selects the i^{th} color based on the *color index* i from the palette and assigns it to the output pixel located at $[x,y]$. Such LUT mapping is compact and can be resolved in constant time $O(1)$. However, this approach cannot be used for an image or video frame that is represented by a number of colors such that $I[0,w] \times [0,h] \rightarrow C \in XYZ$. For these types of *high-color* or *true-color* images, the color mapping process can be formulated as follows:

$$G[x,y] = \tau(I[x,y],P) = \underset{0 \leq i \leq n}{\arg\min} \left(\delta(I[x,y],C_i)\right) \qquad (2)$$

where a distance function $\delta : C_i \times C_j \mapsto \mathbb{R}_+$ computes a distance value between two colors. Thus, the mapping process requires the computation of the minimum color distance for a given color C and all available palette colors in P. Therefore, the overall run-time complexity grows linear with the number n of total palette colors.

With respect to implementing such a non-uniform color quantization for interactive rendering purposes, an approach should adhere to the following requirements:

Real-time Performance (R1): To support color palette mappings for interactive image and video applications, the process should be real-time capable.

Fast Palette Interchange (R2): For flexibility in application, different color palettes should interchangeable easily and fast by requiring only minimal state changes during rendering [1].

Support True-Color Images (R3): The mapping approach should not rely on an indexed color image representation and should support images comprising 8 bit precision or more per color channel.

1.3 Approach and Contributions

To approach the requirements above, we evaluate techniques for mapping an arbitrary input color C_{in} to an output color $C_{out} \in P$ on a per-pixel basis by computing a *minimum color distance* in CIELAB (Lab) color space. By completely preprocessing the minimum color distance search into color LUTs, the mapping process can then be performed in constant time [5] for the sake of increased memory consumptions. Therefore, this paper makes the following contributions:

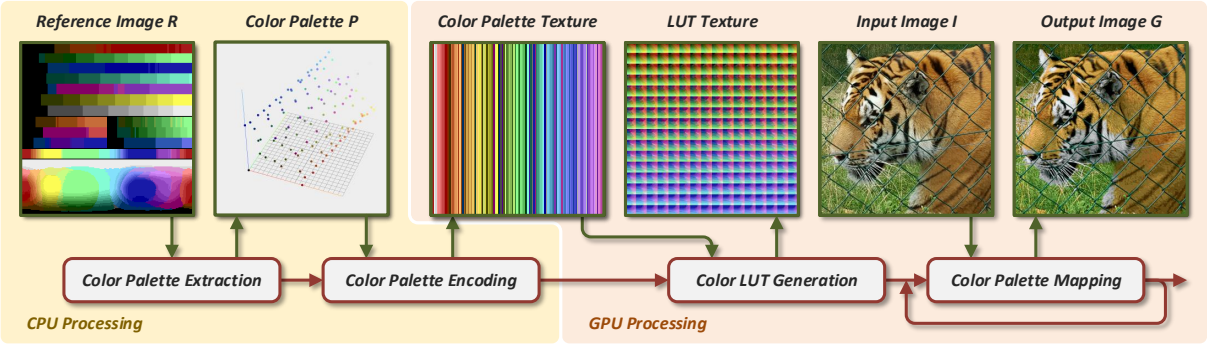


Figure 1: Overview of processing stages and data components as well as control (red) and data flow (green) of the presented real-time palette mapping approach by the example of color reduction using an Atari 800 color palette with 122 colors.

1. It presents approaches to fast color mapping of given color palettes to true-color input images in real-time using GPUs.
2. It provides respective implementation details based on Open Graphics Library (OpenGL) and evaluates the run-time performance.

The remainder of this paper is structured as follows. Section 2 describes the fundamental concept of our approach. Section 3 presents details of GPU-based implementations. Section 4 evaluates and discusses the run-time performance. Section 5 concludes this paper.

2 PALETTE MAPPING CONCEPT

Figure 1 shows an overview of the processing stages, components, and data structures of the GPU-based palette mapping approach. It basically comprises the following steps, covered in the remainder of this section in greater detail:

Color Palette Extraction: As a first (optional) step, a *color palette P* is extracted from a given input *reference image R* (Section 2.1).

Color Palette Encoding: To enable GPU-based processing of the subsequent steps, the extracted color palette P is encoded into a *color palette texture*.

Look-Up Table Generation: To enable an efficient shader-based color mapping implementation, the color palette texture is transformed into a *Look-Up Table (LUT)* (Section 2.3) based on a minimum color distance search (Section 2.2).

Color Palette Mapping: While the previous steps are only required to be performed once per color palette, the color palette mapping use the LUT to implement a point-based operation to map all pixel of multiple input images or video frames I to its respective output representations O using fragment or compute shader programs (Section 3.2).

2.1 Color Palette Extraction

There are basically two approaches for creating a color palette: (1) a user *explicitly* chooses a number of colors or (2) the color palette is *extracted* from a given reference image R.

To support the latter, a tool is created that extracts the color palette from a given image, which is required to be stored without lossy compression, in a preprocessing step. For it, the color of each pixel is added to a set of palette colors resulting in an unordered list of unique colors, which can be stored as image file. This process can be automated and batched for a number of input images. For our purposes, we obtain existing reference images from the Wikipedia encyclopedia (Table 1).

Table 1: Exemplary color palettes comprising different amounts of colors used in this paper.

Palette Name	$\|P\|$	Palette Texture
Teletext / BBC Micro	8	
Apple II	15	
CGA	16	
Commodore 64	16	
Tandy	62	
Atari 800	122	

2.2 Minimum Color Distance Search

For matching two colors C_i and C_j, a distance function $d = \delta(C_i, C_j)$ is computed. We use the Euclidean distance in Lab color space [2] for the color distance computations.

Given an extracted color palette P, the mapping is basically a point-based image processing operation performing a linear minimum color distance search over all its entries for each pixel in the input image (R3, Equation (2), Algorithm 1).

ISBN 978-80-86943-38-1

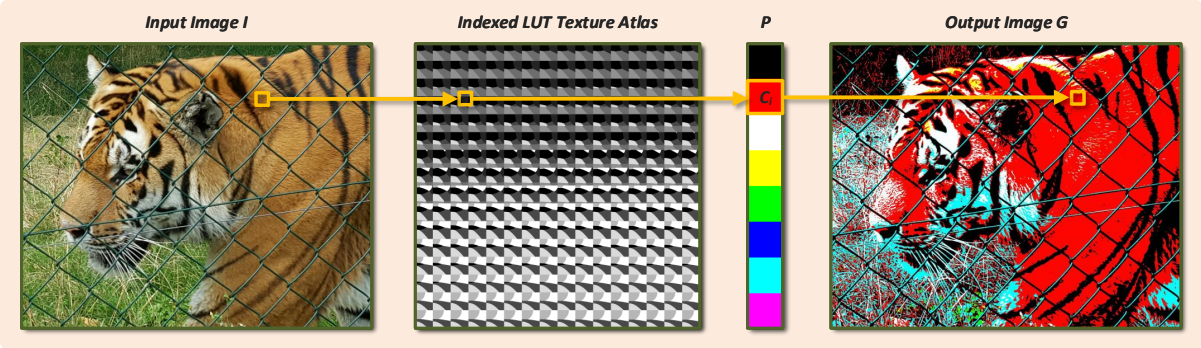


Figure 2: Color quantization using Indexed LUT (ILUT) using a BBC Micro 3 bit palette with 8 colors.

Algorithm 1 Minimum Color Distance Search

Require: $P = C_0, \ldots, C_n$ $n > 0$ {Palette in Lab space}
  for $x = 0$ **to** w **do**
    for $y = 0$ **to** h **do**
      $d_{min} \leftarrow \infty +$ {Initialize}
      $C_{out} \leftarrow [0,0,0]$ {Initialize output color}
      $C_{in} \leftarrow I[x,y]$ {Fetch color value}
      $C_{in}^{Lab} \leftarrow rgb2lab(C_{in})$ {to CIELAB space}
      for $i = 0$ **to** n **do**
        $d = \delta(C_{in}^{Lab}, C_i)$ {Compute color distance}
        if $d < d_{min}$ **then**
          $d_{min} = d$ {Update min. difference}
          $C_{out} \leftarrow C_i$ {Update output color}
        end if
        if $d = 0$ **then**
          break{Early-out of search loop}
        end if
      end for
      $G[x,y] \leftarrow C_{out}$ {Set closest palette color}
    end for
  end for

2.3 Color Look-Up Table Generation

One observation of the minimum color distance search concerns its runtime complexity of $\mathcal{O}(whn)$ for an input image of width w, height h, and a color palette size of n. Considering the high spatial image resolutions of today's cameras, the algorithm's run-time performance easily exceeds the requirement of real-time performance (R1). Thus, the required number of searches is reduced to constant time by using LUTs covering the complete color domain. Similar to the approach described by Selan, a 3D color LUT can be computed in a pre-processing step on a per-palette level [5].

However, in contrast to this approach, we cannot take advantage of using bi-linear texture filtering to reduce the spatial resolution of the LUT. This would potentially introduce colors not represented by the initial palette and yield an imprecise mapping if using nearest-neighbor interpolation. One can distinguish between two LUT variants:

Color LUT (CLUT): This LUT type encodes the mapped palette color for each input color in Red-Green-Blue (RGB) color space (Figure 4(b) and Figure 5(a)). The consumed memory for this representation is $256^3 \cdot 3 \cdot B_C$ with B_C the number of bytes required to represent a color channel. For the palettes used in our applications it is sufficient to use 8 bit channel precision resulting in a Video Random Access Memory (VRAM) footprint of 50 MB

Indexed LUT (ILUT): Instead of storing the mapped color values directly (Equation (1)), the memory footprint can be reduced by storing only the respective color indexes (Figure 5(b)). Thus, the space-complexity reduces to $4096^2 \cdot B_I + |P| \cdot 3 \cdot B_C$, with B_I the number of bytes required per index. Usually, it is sufficient to use an 8 bit index precision, resulting in a VRAM memory footprint of 16 MB in addition to the color palette. Figure 2 shows an overview of the color mapping process using an ILUT.

3 GPU-BASED IMPLEMENTATION

The concept of the previous section is prototypical implemented using OpenGL [4] and OpenGL Shading Language (GLSL). However, these implementations can be easily transferred to other graphics Application Programming Interface (API) as well as Embedded System (ES) variants.

The remainder of this section briefly describes how palettes and different LUT types are represented on GPUs, and how the particular color mapping processes are implemented by the example of using OpenGL fragment shader programs. Such implementations are required because rendering using *paletted textures* are not supported by OpenGL and respective legacy formats are declared deprecated since OpenGL 3.0. Further, the support for the `GL_EXT_paletted_texture` extension has been dropped by the major hardware vendors.

3.1 GPU-based Palette Representation

Before describing the different palette mapping implementation, this section briefly present the palette and LUT representations suitable for GPU-based rendering.

ISBN 978-80-86943-38-1

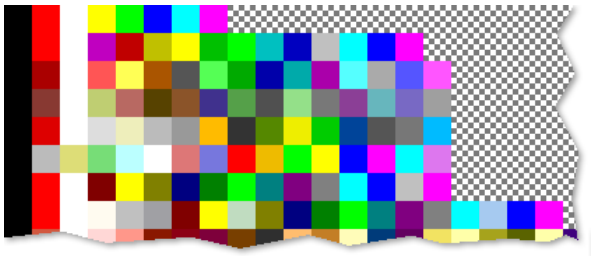


Figure 3: Close-up of an exemplary 2D TA that stores multiple color palettes row-wise.

3.1.1 Texture-based Palette Representation

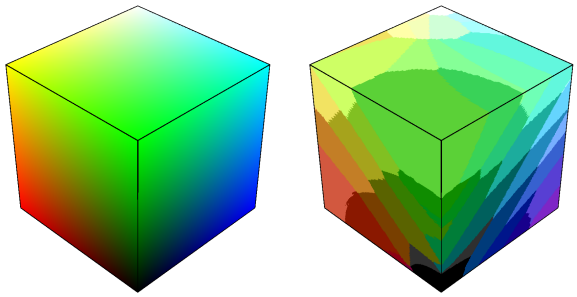
In general, each palette is represented using a single 2D texture. During runtime, a palette is converted into Lab color space [2] and uploaded to a *palette texture* within VRAM. The straightforward approach to standard palette representation for GPUs is using a three channel 1D texture, i.e., a 2D texture with fixed height of 1 pixel. Table 1 shows some texture examples. This approach provides optimal texture utilization.

However, the total amount of palette colors is limited to the maximum texture size. In rare cases that this size is exceeded, the palette is wrapped and indexing is performed by modulo operation. To minimize state changes during rendering, which can be introduced when switching palette textures, and to facilitate fast palette interchanges, such texture representation can be combined with texture batching [6]. This enables the storage of multiple palettes using a 2D Texture Atlas (TA) (Figure 3).

3.1.2 Look-up Table Representations

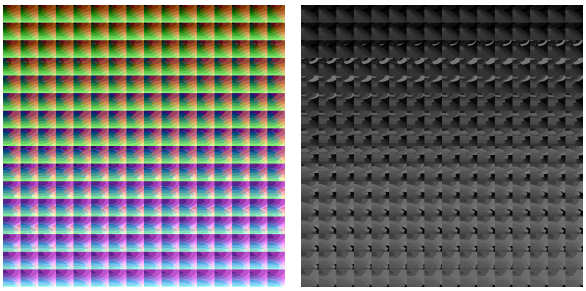
There are basically two texture-based representations of CLUTs and ILUTs for GPU-based implementation that take advantages of build-in texturing functionality of shading languages:

3D Textures: An effective way to represent a LUT is using a 3D texture of size 256^3 texels for precise mapping of low precision color representations (Figure 4). 3D textures are also supported for mobile devices since OpenGL ES 3.0 and thus are commonly available.



(a) *Identity Palette.* (b) *Quantized Palette.*

Figure 4: 3D texture LUTs.



(a) *CLUT TA.* (b) *ILUT TA.*

Figure 5: Comparison of 2D texture atlas LUT representations; each of 4096^2 pixels spatial resolution.

2D Texture-Atlases: Figure 5 shows a comparison between a TA containing all 256 layer of a CLUT (Figure 5(a)) and ILUT (Figure 5(b)). Such an atlas can be used in combination with 2D texture arrays to support fast interchanges of palettes (R2).

The texture format used for CLUTs is `GL_RGB` with an internal `GL_UNSIGNED_BYTE` representation is sufficient. For ILUT, `GL_LUMINANCE` using `GL_FLOAT` internal format at a minimum of 16 bit precision to represent the range of possible palette indexes for $n > 256$.

3.2 Shader-based Color Quantization

The shader-based color quantization can be performed within a single rendering pass. Therefor, a Screen-aligned Quad (SAQ) that is textured with the input image covering the complete viewport is rasterized with an activated fragment shader program and bound texture resources. The remainder of this section shows the particular fragment shaders for performing the color mapping operations for the respective texture representations described previously.

3.2.1 Minimum Color Distance Search

For texture-based palette representations, Listing 1 shows a GLSL implementation for the minimum color distance search (Section 2.1). Performed run-time tests comparing this implementation with variants that support early-out of the for loop (Algorithm 1) that, however, yield inferior performance presumably due to coherency issues [3].

3.2.2 Sampling Look-Up Tables

For sampling LUT textures, the input color values $C = (r,g,b) = (s,t,r) = I[x,y] \in [0,1]^3$ fetched at the respective fragment coordinates $(x,y) \in [0,0] \times [w,h]$ are used. While sampling 3D textures is natively supported by GLSL language features, LUTs that are represented by 2D TAs require additional instructions to resolve the indirection. Listing 2 shows a GLSL function for sampling LUTs represented as a 2D TA.

ISBN 978-80-86943-38-1

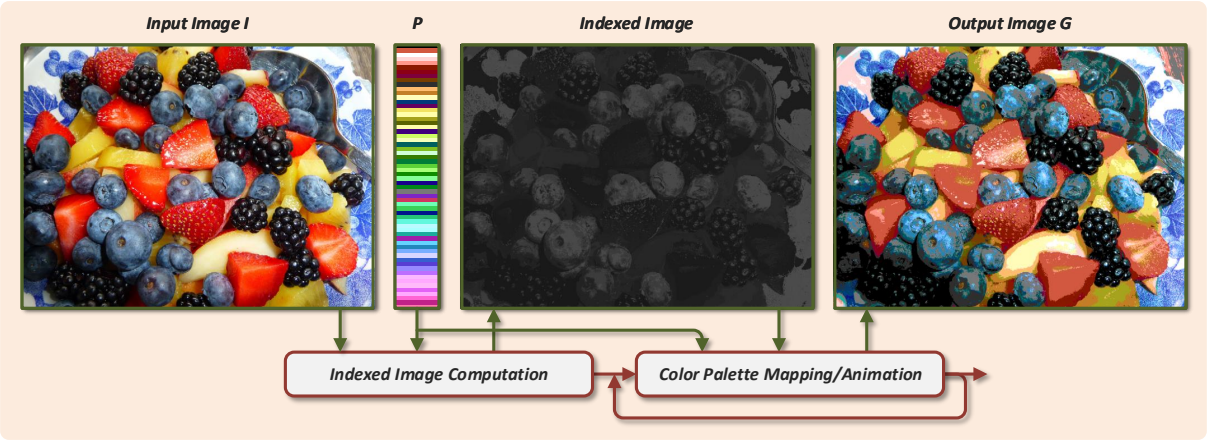


Figure 6: Overview of indexed image computation and subsequent color quantization by the example of an Atari 800 color palette with 122 colors.

```
int closestColorIndex(
  const in vec3 sampleLab, // Color in Lab
  const in sampler2D P,    // Palette TA in Lab
  const in int index)      // Palete to use
{
  // Determine palette length for palette index
  int pSize = texelFetch(P, ivec2(0, index), 0).x;
  int colorIndex = 0;
  float minimumDistance = FLT_MAX;
  // Linear search in palette
  for(int i = 1; i < pSize; i++) {
    // Compute color distance
    float d = distance(
      texelFetch(P, ivec2(i, index), 0).rgb,
        sampleLAB);
    // Minimum distance compare
    if(d < minimumDistance) {
      minimumDistance = d;
      colorIndex = i;
    }
  }
  return colorIndex; // Index of closest color.
}
```

Listing 1: GLSL implementation of minimum color distance search.

```
vec2 offset(const in float s, const in vec4 info){
  return info.zw * vec2(mod(s, info.y),
    floor(s * info.z));
}

vec4 sampleAs3DTexture(
  const in sampler2D LUT,
  const in vec3 texCoord,
  const in vec4 info) {

  float sliceZ = floor(texCoord.z * info.x);

  vec2 slice0Offset = offset(sliceZ, info);
  vec2 slicePixelSize = info.zw / info.x;
  vec2 uv = slicePixelSize * 0.5 + texCoord.xy *
            (info.zw - slicePixelSize);
  return texture2D(tex, slice0Offset + uv);
}
```

Listing 2: GLSL implementation for sampling a 2D texture atlas using given 3D texture coordinates.

3.2.3 *Quantization by Color Look-Up Tables*

Once, the CLUTs are generated, the actual quantization can be efficiently performed by sampling these using the RGB color values as texture coordinates similar to the approach described by Selan [5]. Listing 3 shows a GLSL function for color quantization using an CLUT requiring a single texture look-up.

```
vec4 mapColorCLUT(
  const in vec3 color,     // color to map
  const in sampler3D CLUT) // look-up table
{
  return texelFetch(CLUT, ivec3(color), 0);
}
```

Listing 3: GLSL implementation of CLUT mapping.

3.2.4 *Quantization by Indexed Look-Up Tables*

Compared to the CLUT-based quantization, ILUT-based quantization requires an additional sampling operation to resolve the indirection between ILUT and the color palette. Listing 4 shows a GLSL implementation for this mapping operation.

```
vec4 mapColorILUT(
  const in vec3 color,        // Color to map
  const in sampler2D palette, // Palette texture
  const in sampler3D ILUT)    // Indexed LUT
{
  // Obtain index into color palette
  int index = texelFetch(ILUT, ivec3(color), 0).r;
  // Sampling color palette
  return texelFetch(palette, ivec2(index, 0), 0);
}
```

Listing 4: GLSL implementation of ILUT mapping.

3.3 LUT and Indexed Image Generation

The task of LUT generation can be efficiently performed on GPU, for both 3D texture and 2D TAs using compute shader or off-screen rendering in combination with Frame-Buffer Object (FBO) and

ISBN 978-80-86943-38-1

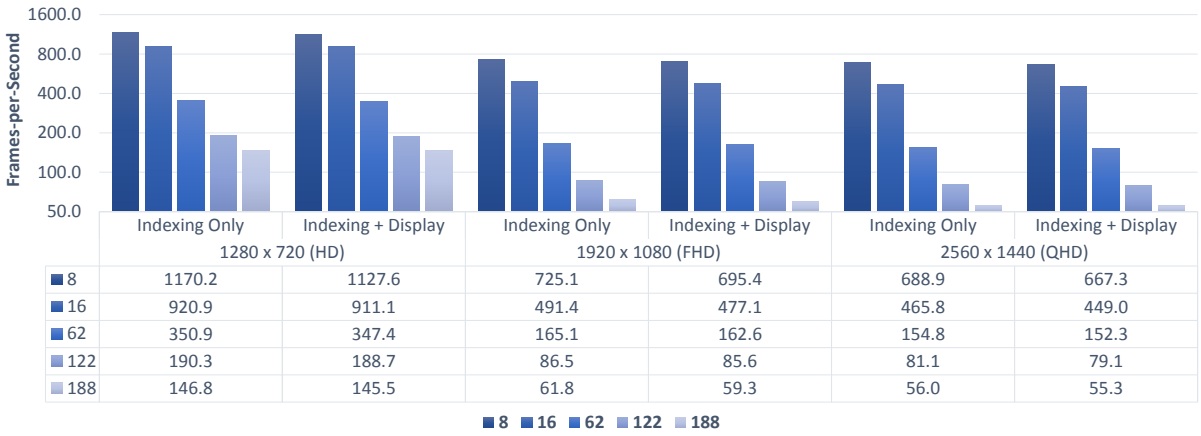


Figure 7: Performance comparison of GPU-based LUT generation (indexing) and combined mapping (display) in Frames-per-Second (FPS) (logarithmic scale) for color palettes of different complexity (8 to 122) and different input image resolutions.

fragment shader support (can be optimized by `GL_EXT_shader_framebuffer_fetch`). Both are based on the minimum distance search implementation described in Section 3.2.1. In general, these techniques can be used to "bake" any kind of LUT. The overall goal is avoid color space conversion at mapping time by using RGB color space values only.

The 2D TAs processing approach can be used the generate an intermediate *indexed image* representation, that can be used for rendering animated palettes for mimicking *palette cycling*. Therefore, the input image I is transformed once to an indexed image G using the implementation of Section 3.2.1. Palette animations can be easily achieved algorithmically or represented using a 2D TA of palette texture key-frames (Figure 6).

4 PERFORMANCE EVALUATION

This section presents performance evaluations of all three approaches described before. In addition thereto, the run-time performance for color mapping and the respective memory footprints of the particular data representations are compared.

4.1 Datasets and Test Systems

Different image resolutions were tested to estimate the run-time performance regarding the spatial resolution of an image as well as different palette complexity (Table 1). The following common resolutions were chosen: 1280×720 (HD), 1920×1080 (FHD), and 2560×1440 (QHD) pixels. We tested the rendering performance of our preliminary implementation was conducted using a NVIDIA GeForce GTX 970 GPU with 4 096 MB VRAM on a Intel Xeon CPU with 2.8 GHz and 12 GB RAM. Rendering was performed in windowed mode with vertical synchronization turned off.

The measurements in FPS are obtained by averaging 500 consecutive frames. For all off-screen rendering passes, fragment shader functionality using a textured SAQ with a geometric complexity of four vertices and two triangle primitives. For rasterization, back-face culling [1] is enabled and depth test and writing to depth buffer are disabled.

4.2 Performance Results

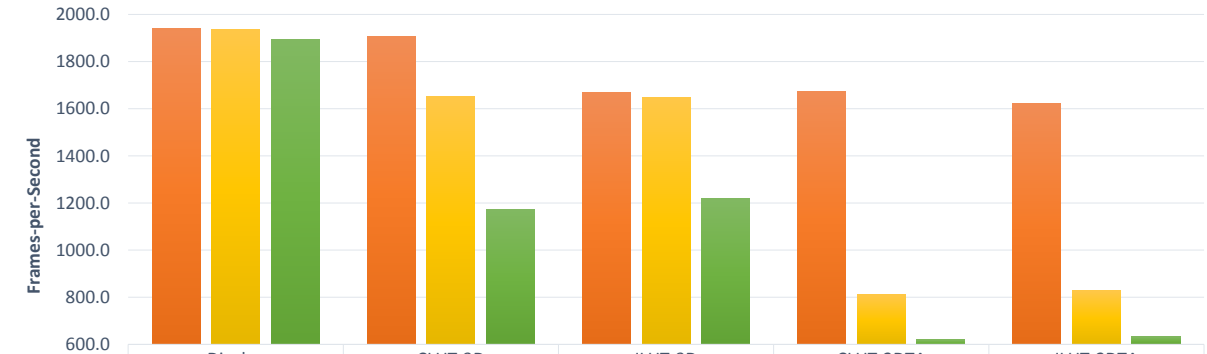
Figure 7 shows the performance evaluation for the minimum color distance search implementation (Section 3.2.1) for different color palette complexities (8 to 188 colors) by comparing the computation of an indexed image representation G only (*indexing*) as well as in combination with a subsequent display pass that applies the mapping implementation of Section 3.3.

It can be observed, that run-time performance of indexing decreases linearly with increasing color palette complexity, while the performance impact of color mapping during display is neglectable. The overall performance decrease with respect to increasing input image resolution indicates that the technique is fill-limited.

Figure 8 shows the results of the run-time performance evaluation in FPS of the different GPU-based LUT representations (CLUT and ILUT using 3D textures and Texture Atlass (TAs)), algorithms (Section 3.2.3, and Section 3.2.4) impacting the color mapping performance compared to a pass-through display pass.

It can be observed that 3D texture representations for CLUT and ILUT are superior over 2D TAs, especially for high spatial input resolutions. This can be explained by the additional instructions required to transform the texture coordinates to sample 2D TAs and probable texture-cache violations. Independent of their representations are ILUTs inferior to CLUTs. This can be explained by the additional texture sampling operation required to resolve the palette indirection, which most probably contributes to additional texture-cache viola-

 ISBN 978-80-86943-38-1



	Display	CLUT-3D	ILUT-3D	CLUT-2DTA	ILUT-2DTA
1280 x 720	1938.5	1907.0	1667.9	1673.3	1623.7
1920 x 1080	1935.7	1651.6	1649.1	810.3	828.3
2560 x 1440	1893.3	1174.2	1218.5	621.2	633.9

Figure 8: Performance of GPU-based LUT mapping performance in FPS for different LUT representations compared to a pass-through display pass.

tions. These observations show all characteristics of the traditional space-time/time-memory trade-off.

5 CONCLUSIONS

This paper presents and discusses an approach for fast real-time color palette quantization for true-color input images. The palette quantization is based on computing minimal color difference using the Lab color space. It is shown, that this mapping can be performed in real-time and can be significantly optimized by using LUTs computed in a pre-processing step on GPUs. Depending on the use-case, the described approaches allows for trading run-time performance for VRAM memory consumption.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable feedback. This work was funded by the Federal Ministry of Education and Research (BMBF), Germany, for the AVA project 01IS15041.

REFERENCES

[1] Tomas Akenine-Möller, Eric Haines, Naty Hoffman, Angelo Pesce, Michał Iwanicki, and Sébastien Hillaire. *Real-Time Rendering 4th Edition*. A K Peters/CRC Press, Boca Raton, FL, USA, 2018.

[2] B. Hill, Th. Roger, and F. W. Vorhagen. Comparative analysis of the quantization of color spaces on the basis of the cielab color-difference formula. *ACM Trans. Graph.*, 16(2):109–154, April 1997.

[3] Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, and Randi Rost. Real-time shading. In *ACM SIGGRAPH 2004 Course Notes*, SIGGRAPH '04, New York, NY, USA, 2004. ACM.

[4] Mark Segal and Kurt Akeley. *The OpenGL Graphics System: A Specification Version 4.5*. Silicon Graphics Inc., 4.5 edition, June 2017.

[5] Jeremy Selan. Using Lookup Tables to Accelerate Color Transformations. In *GPU Gems*, pages 381–392. Addison-Wesley, 2004.

[6] Matthias Wloka. *ShaderX3*, chapter Improved Batching via Texture Atlases, pages 155–167. Charles River Media, 2005.

ISBN 978-80-86943-38-1

ISBN 978-80-86943-38-1

ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 2901
WSCG Proceedings Part II

The Role of Polarization in The Multiple Reflections Modeling

Alexander Y. Basov
Light Engineering Department Moscow Power Engineering Institute, Moscow Russia
BasovAlY@mpei.ru

Vladimir P. Budak
Light Engineering Department Moscow Power Engineering Institute, Moscow Russia
BudakVP@gmail.com

Anton V. Grimailo
Light Engineering Department Moscow Power Engineering Institute, Moscow Russia
GrimailoAV@gmail.com

ABSTRACT

This article shows a new mathematical model for calculation of multiple reflections based on the Stokes vector and Mueller matrices. The global illumination equation and local estimations method were generalized on the polarization case. The results of the calculation of multiple reflections using the local estimations method show a difference of more than 30% between the standard calculation and the polarization-accounting one. A way to describe the surface reflection with polarization account is proposed.

Keywords

Polarization, global illumination equation, local estimations, Stokes vector, Mueller matrix, Monte-Carlo.

1. INTRODUCTION

In recent years, visualization of the light distribution obtained by using a lighting device has become an integral part of the design of lighting systems. Thus, the calculation of multiple reflections has become exceedingly important.

Traditionally, the state of light polarization is not considered in the calculations. Similarly, lighting modeling software (e. g. the industry standard software DIALux and Relux) do not consider the influence of the light polarization in the calculation of light distribution.

This neglection is acceptable when we work with diffusely reflecting surfaces and a small number of re-reflections. On the contrary, the influence of light polarization must be considered by surfaces with a significant specular part. The very first reflection changes the state of light polarization (even if the light has been depolarized totally before the action) and this fact will affect all the following processes of light distribution.

It is evident that after a series of reflections the light becomes depolarized again. However, the effect of the light polarization on the quantitative results of the calculation remains unknown. By current estimates [Mishchenko et al. 1994], the difference between results of the standard calculation and the calculation considering the polarization may reach more than 20%. This difference may be even more significant if we use the polarization-based model of reflecting surface which considers the light scattering in a material volume.

To date, a series of proceedings devoted to the polarization account in visualization problems have been published (e.g. [Mojzik et al. 2016]). They show that polarization state accounting leads to quite significant changes in the pictures obtained during visualization. However, despite of the abundance of research of such kind, the question of the light polarization influence on the quantitative characteristics of the light distribution has not been considered yet. At the same time, it is these characteristics that primarily interest light planners when they solve practical problems.

Thus, we decided to study the influence of the light polarization on the quantitative result of the calculation of multiple reflections. To achieve this goal, we need to solve the following tasks.

1. Pass from the wave description of the light to the ray approximation as the latter is more in nature of the phenomena considered.
2. Bring the global illumination equation for the Stokes vector to the volume integration. Existing equation based on surface integration [Mojzik et al. 2016] does not allow constructing mathematically rigorous ray-based algorithms.

3. Calculate multiple reflections using local estimations of the Monte-Carlo method (as they appear to be the most efficient for solving the problem) and evaluate the influence of the light polarization account on the quantitative result.

We are also confident there is a need for a mathematical model of reflecting surface describing the light reflection not only from the face of material but from the volume of material as well. In this case, the role of the light polarization is highly significant too.

2. MATHEMATICAL MODEL OF MULTIPLE REFLECTIONS WITH POLARIZATION ACCOUNT

Description of polarization in the electromagnetic field theory using ellipses is completely unknown to the description of radiation in ray optics which is based on the reaction of the optical radiation receiver. The most suitable way to develop the mathematical model with polarization-accounting parameters would be to use the Stokes vector (for describing the ray parameters) and 4×4 Mueller matrices [Mueller H. 1943] (for describing the parameters of surface reflection).

The Stokes vector $\mathbf{L}$ components are determined by the quadratic receiver reactions J_i as

$$L_0 = 2J_0, \quad L_1 = 2(J_0 - J_1), \\ L_2 = 2(J_2 - J_0), \quad L_3 = 2(J_3 - J_0), \tag{1}$$

where specific polarization filters differ $J_i, i \in \overline{0,3}$: J_0 is for the neutral filter with the transmittance $\tau = 0.5$; J_1 is for the reference plane (the "system of readout") defining analyzer; J_2 is for the analyzer with the axis at the angle 45° to the reference plane; J_3 is for complex filter of quarter-wave plate and analyzer at the angle 45° to the reference plane.

The result of the interaction of the surface and the ray described with the Stokes vector can be written as:

$$\mathbf{L}(\mathbf{r},\hat{\mathbf{l}}) = \ddot{\rho}(\mathbf{r},\hat{\mathbf{l}},\hat{\mathbf{l}}')\mathbf{L}(\mathbf{r},\hat{\mathbf{l}}) \tag{2}$$

where $\mathbf{L}(\mathbf{r},\hat{\mathbf{l}}')$ and $\mathbf{L}(\mathbf{r},\hat{\mathbf{l}})$ are the radiances before and after the interaction respectively; $\ddot{\rho}(\mathbf{r},\hat{\mathbf{l}},\hat{\mathbf{l}}')$ is the Mueller matrix of the surface (we consider in this article the reflection only). Expression (2) is correct when the reference planes of the incident are the same as the reflected rays. In general case, we must account the rotation of the reference system with a special matrix. The matrix which is the multiplication of each transformation of the Mueller matrix provides the successive transformations result.

Note that hereinafter we use the following notation:

$\mathbf{a}$ — column vector; $\bar{\mathbf{a}}$ — row vector; $\hat{\mathbf{a}}$ — unit column vector; $\ddot{\mathbf{a}}$ — matrix.

The questions of simulation of lighting systems and visualization of 3D scenes in computer graphics are based on finding the solution of the global illumination equation [Kajiya J. T. 1986]

$$L(\mathbf{r},\hat{\mathbf{l}}) = L_0(\mathbf{r},\hat{\mathbf{l}}) + \\ + \frac{1}{\pi}\int L(\mathbf{r},\hat{\mathbf{l}}')\sigma(\mathbf{r},\hat{\mathbf{l}},\hat{\mathbf{l}}')F(\mathbf{r},\mathbf{r}')\Theta(\mathbf{r},\mathbf{r}')d^2\mathbf{r}', \tag{3}$$

$$F(\mathbf{r},\mathbf{r}') = \frac{|(\hat{\mathbf{N}},\hat{\mathbf{l}}')(\hat{\mathbf{N}}',\hat{\mathbf{l}}')|}{(\mathbf{r}-\mathbf{r}')^2} = \frac{|(\hat{\mathbf{N}},\mathbf{r}-\mathbf{r}')(\hat{\mathbf{N}}',\mathbf{r}-\mathbf{r}')|}{(\mathbf{r}-\mathbf{r}')^4},$$

where $L(\mathbf{r},\hat{\mathbf{l}})$ is the radiance at the point $\mathbf{r}$ in the direction $\hat{\mathbf{l}}$, $\sigma(\mathbf{r},\hat{\mathbf{l}},\hat{\mathbf{l}}')$ is used for the bidirectional scattering distribution function (reflection or transmission), $L_0(\mathbf{r},\hat{\mathbf{l}})$ is the direct radiation radiance near the sources, $\hat{\mathbf{N}}$ is the scene surface normal vector, $\Theta(\mathbf{r},\mathbf{r}')$ is the function of the surface element $d^2\mathbf{r}'$ visibility from the point $\mathbf{r}$.

Thus, guiding the considerations used in derivation (3) one can obtain an analogous equation for the Stokes vector:

$$\mathbf{L}(\mathbf{r},\hat{\mathbf{l}}) = \mathbf{L}_0(\mathbf{r},\hat{\mathbf{l}}) + \\ + \frac{1}{\pi}\oint \ddot{\mathrm{R}}(\hat{\mathbf{l}}'\times\hat{\mathbf{l}} \to \hat{\mathbf{N}}\times\hat{\mathbf{l}})\ddot{\rho}(\mathbf{r},\hat{\mathbf{l}},\hat{\mathbf{l}}')\times \\ \times \ddot{\mathrm{R}}(\hat{\mathbf{N}}\times\hat{\mathbf{l}}' \to \hat{\mathbf{l}}'\times\hat{\mathbf{l}})\mathbf{L}(\mathbf{r},\hat{\mathbf{l}}')F(\mathbf{r},\mathbf{r}')\Theta(\mathbf{r},\mathbf{r}')d^2\mathbf{r}', \tag{4}$$

where $\mathbf{L}(\mathbf{r},\hat{\mathbf{l}})$ is the Stokes vector; $\ddot{\mathrm{R}}(\hat{\mathbf{l}}'\times\hat{\mathbf{l}} \to \hat{\mathbf{N}}\times\hat{\mathbf{l}})$ is the reference plane rotation matrix; $\ddot{\rho}(\mathbf{r},\hat{\mathbf{l}},\hat{\mathbf{l}}')$ is the Mueller matrix of reflectance.

In general, the global lighting equation has no analytical solution. For this reason, the numerical methods are used mainly for solving it. The most popular approach is to use the Monte-Carlo Methods, as they have shown to be efficient comparing to other methods.

The algorithm of calculation of lighting systems and visualization of 3D scenes based on local estimations of the Monte-Carlo Method [Kalos M. 1963] seems to be the most advanced in its class. This approach enables to speed up the calculations by 80-90 times compared to direct modeling [Budak et al. 2012]. This is especially important for polarization-accounting calculations as they require much more operations on each step of the algorithm.

The general meaning of the transition to the local estimations is the transformation of the surface integration in (4) into the integration over the volume by using δ-function. This enables the development of a ray-based modeling algorithm.

ISBN 978-80-86943-38-1

In order to enable the transformation we need to account that $\mathbf{r}'$ and $\hat{\mathbf{l}}'$ are connected by the following expression:

$$\hat{\mathbf{l}}' = \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|}. \quad (5)$$

Then the equation (4) takes the form:

$$\mathbf{L}(\mathbf{r}, \hat{\mathbf{l}}) = \mathbf{L}_0(\mathbf{r}, \hat{\mathbf{l}}) +$$
$$+\frac{1}{\pi}\int \vec{\mathrm{R}}(\hat{\mathbf{l}}' \times \hat{\mathbf{l}} \to \hat{\mathbf{N}} \times \hat{\mathbf{l}})\vec{\rho}(\mathbf{r}, \hat{\mathbf{l}}, \hat{\mathbf{l}}')\vec{\mathrm{R}}(\hat{\mathbf{N}} \times \hat{\mathbf{l}}' \to \hat{\mathbf{l}}' \times \hat{\mathbf{l}}) \times \quad (6)$$
$$\times \delta(\hat{\mathbf{l}} - \hat{\mathbf{l}}')\mathbf{L}(\mathbf{r}, \hat{\mathbf{l}}')F(\mathbf{r}, \mathbf{r}')\Theta(\mathbf{r}, \mathbf{r}')d^3\mathbf{r}'.$$

Now, one can show the integral in (6) as a Neumann series and after some transformations [Marchuk G. I. 1980] get an expression for local estimation of Stokes vector at the point $\mathbf{r}$:

$$\mathbf{I}_\varphi = \mathrm{M}\sum_{n=0}^{\infty}\vec{k}(\mathbf{r}, \mathbf{r}')\mathbf{Q}_n, \quad (7)$$

$$\vec{k}(\mathbf{r}, \mathbf{r}') = \vec{\mathrm{R}}(\hat{\mathbf{l}}' \times \hat{\mathbf{l}} \to \hat{\mathbf{N}} \times \hat{\mathbf{l}})\vec{\rho}(\mathbf{r}, \hat{\mathbf{l}}, \hat{\mathbf{l}}')\vec{\mathrm{R}}(\hat{\mathbf{N}} \times \hat{\mathbf{l}}' \to \hat{\mathbf{l}}' \times \hat{\mathbf{l}}) \times$$
$$\times F(\mathbf{r}, \mathbf{r}')\Theta(\mathbf{r}, \mathbf{r}'),$$

where $\mathbf{Q}_n$ is the vector weight of the ray, which components correspond to those of the Stokes vector.

3. THE MODEL IMPLEMENTATION AND OBTAINED RESULTS

The mathematical model described above was implemented using the numerical-computing environment MATLAB. In order to make the algorithm simpler and reduce the calculation time, the authors decided to assume the diffuse distribution of the rays reflected from the surface. This way enables to estimate the first approximation of the influence of polarization account in the multiple reflection calculations.

A 1×1×1 cube "room" was chosen as a modeling scene. There are points of illuminance estimation on the "floor" and a Lambertian luminous disc (diameter 0.1) in the center of the "ceiling" (Figure 1).

The sum of two matrices was used as a reflection matrix when calculating:

$$\vec{\rho} = a\vec{\rho}_F + (1 - a)\vec{\rho}_L,$$

where $\vec{\rho}_F$ is Fresnel reflection (Mueller) matrix; $\vec{\rho}_L$ is Lambertian reflection matrix; a is Fresnel part in reflection $(0 < a < 1)$. Matrix $\vec{\rho}_L$ is a zero-matrix with an only non-zero element ρ_{L11} for the reflection coefficient.

The model surfaces have the following parameters: reflection coefficient — 0.5; refractive index — 1.5; a changes from 1 to 0. Figure 2 demonstrates the results normalized relative to the number of rays emitted from the disc.

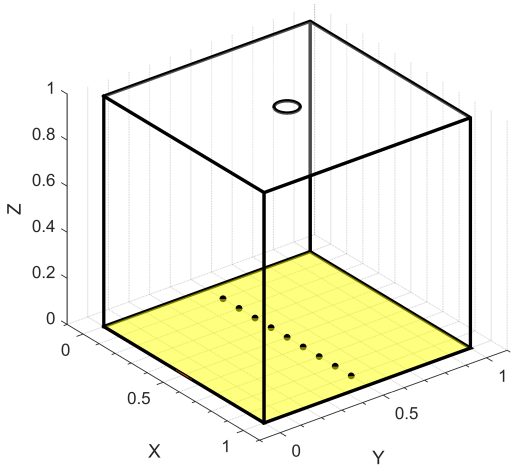


Figure 1. The modeling scene.

One can see, that already at a = 0.2 the difference between the values of illuminance is about 10% and at a = 0.6 more than 20%.

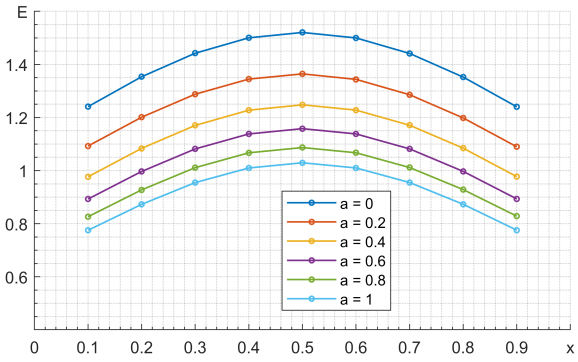


Figure 2. The results of the multiple reflections modeling (relative units).

4. FURTHER DEVELOPMENT OF THE MODEL

When calculating multiple reflections, we accept the assumption that the reflected ray diffuse distribution in order to account the light reflection not only from the material surface but also from the material volume. The light penetrates the near-surface layers of the material where the light scattering by the material particles occurs. Then a certain fraction of the initial luminous flux re-enters the surrounding space. At this point, the role of the light polarization is highly significant too.

The light scattering by the material particles in the reflecting layer is inherently similar to the radiative transfer in turbid media. Therefore, at the first stage of model development, we are going to represent the reflecting surface as a scattering layer with diffusely reflecting plane at the bottom and randomly rough Fresnel boundary above.

ISBN 978-80-86943-38-1

In the general case, the scattering media is characterized by matrix scatter coefficient $\vec{\sigma}$, matrix absorption coefficient $\vec{\kappa}$ and matrix extinction coefficient $\vec{\varepsilon} = \vec{\kappa} + \vec{\sigma}$. Neglection of dichroism, birefringence and similar effects which are inherent only for several materials enables the transition to scalar analogs of matrix coefficients $\varepsilon=\kappa+\sigma$.

Thus, for modeling of surface reflection with the assumptions above we need to solve the boundary value problem for the vector radiative transfer equation. Consider the plane-parallel layer with the diffuse bottom. The layer is irradiated by the plane monodirectional source with a random polarization state. Then one can write the problem as

$$\begin{cases} \mu \dfrac{\partial}{\partial \tau}\mathbf{L}(\tau,\hat{\mathbf{l}}) + \mathbf{L}(\tau,\hat{\mathbf{l}}) = \dfrac{\Lambda}{4\pi}\oint \vec{\mathrm{R}}(\hat{\mathbf{l}}\times\hat{\mathbf{l}}' \to \hat{\mathbf{N}}\times\hat{\mathbf{l}})\times \\ \times \vec{x}(\tau,\hat{\mathbf{l}},\hat{\mathbf{l}}')\vec{\mathrm{R}}(\hat{\mathbf{N}}\times\hat{\mathbf{l}}' \to \hat{\mathbf{l}}\times\hat{\mathbf{l}}')\mathbf{L}(\tau,\hat{\mathbf{l}})d\hat{\mathbf{l}}'; \\ \mathbf{L}(0,\mu>0,\varphi) = \mathbf{L}\delta(\hat{\mathbf{l}}-\hat{\mathbf{l}}'); \\ \mathbf{L}(\tau_0,\mu\le 0,\varphi) = [\rho E/\pi \quad 0 \quad 0 \quad 0]^T, \end{cases} \tag{8}$$

where $\mu = \cos\theta$; $\tau = \int_{z_0}^{z_1} \varepsilon(z)dz$ is the optical track thickness in the section $[z_0, z_1]$; $\Lambda = \sigma/\varepsilon$ is the single scatter albedo; $\vec{x}(\tau,\hat{\mathbf{l}},\hat{\mathbf{l}}')$ is the scatter matrix.

The randomly rough Fresnel boundary modeling is a nontrivial problem as well. There are two approaches to solving this problem [Kargin B. A. 2000]. In the first one, the random field realizations are constructed according to the randomization principle. Then the random trajectories are simulated for the obtained realizations and on their basis estimations for functionals are calculated. However, this requires to find the trajectory and surface intersection points and thus, much computational time.

Therefore, the second approach based on the method of mathematical expectations is preferable. For the construction of a random *N*-trajectory here, realizations of surface elevations are constructed only at *N*-points computed in a certain way. At the points of photon scattering on the random surface, the selection is made random realizations of normals to the surface.

5. CONCLUSION

It is obvious now that consideration of the light polarization leads to significant changes in the quantitative results of the calculation of multiple reflections. Moreover, the proposed mathematical model remains within the framework of the standard photometrical concepts but generalizes them to the polarization case. The luminance is a vector value and reflection coefficient transforms into a matrix. The reference plane rotation should be also accounted.

The global illumination equation was generalized on the polarization case and brought to the volume integration as well to mathematically rigorously formulate calculation algorithms with polarization accounting.

Another result of the study is the introduction of light polarization in the algorithm based on the local estimations. This method allows estimating illuminance at the specific points of interest and reduces the calculation time by 80-90 times.

We are also confident that further development of the model by using the reflecting surface representation described above will enable obtaining more precise results of the light distribution simulation.

6. ACKNOWLEDGMENTS

We want to thank the ILEC BL Group and personally its president Boos G. V. for help they provide to us.

7. REFERENCES

[Budak et al. 2012] Budak, V.P., Zheltov, V.S., Kalatutsky, T.K. Local estimations of Monte Carlo method with the object spectral representation in the solution of global illumination. Computer Research and Modeling. 2012, Vol. 4, N.1, p.75-84, 2012

[Kajiya J. T. 1986] Kajiya, J.T. The rendering equation. In Proceedings of SIGGRAPH 1986. V.20, N4. – P.143-150, 1986

[Kalos M. 1963] Kalos, M.H. On the Estimation of Flux at a Point by Monte Carlo. Nuclear Science and Engineering. 1963, Vol. 16, N.1, p.111-117, 1963.

[Kargin B. A. 2000] Kargin B.A. Statistical modeling of stochastic problems of the atmosphere and ocean optics. Proc. SPIE 4341, Seventh International Symposium on Atmospheric and Ocean Optics, (29 December 2000).

[Marchuk G. I. 1980] Marchuk, G.I. Monte-Carlo Methods in Atmospheric Optics. Berlin: Springer-Verlag, 1980.

[Mishchenko et al. 1994] Mishchenko, M.I., Lacis, A.A., Travis, L.D. Errors induces be the neglect of polarization in radiance calculations for Rayleigh-scattering atmospheres. Journal of Quantitative Spectroscopy and Radiative Transfer. 1994, Vol. 51, No. 3, pp. 491-510, 1994

[Mojzik et al. 2016] Mojzik, M., Skrivan, T., Wilkie, A., Krivanek, J. Bi-Directional Polarised Light Transport. Eurographics Symposium on Rendering 2016.

[Mueller H. 1943] Mueller, H. Memorandum on the Polarization Optics of the Photo-Elastic Shutter. Report Number 2 of the OSRD Project OEMsr, Boston, MA, USA, 1943; p. 57

 ISBN 978-80-86943-38-1