

Project Documentation

Interactive Lighting Detector

written by

Vera Brockmeyer (Matrikelnr. 11077082)
Laura Anger (Matrikelnr. 11086356)

Image Processing in SS 2017

Supervisor:

Prof. Dr. Dietmar Kunz
Institute for Media- and Phototechnology

Contents

1	Introduction	4
1.1	Motivation	4
1.2	Usage Context	4
1.3	Project Goal	4
2	State of the Art	5
2.1	Image Forensic	5
2.2	Light Vectors	5
3	Materials	6
3.1	Hardware	6
3.2	Software	6
3.2.1	QT	6
3.2.2	OpenCV	6
3.3	Testimages	7
3.3.1	First Batch	7
3.3.2	Second Batch	8
4	System	9
4.1	Lighting Model	9
4.2	Contours	9
4.2.1	Find Contours	9
4.3	Subcontours	10
4.4	Different Approaches	10
4.4.1	1. Approach: One Lightvector	10
4.4.2	2. Approach: Averaging Lightvectors	10
4.4.3	3. Approach: Lightvector with highest Intensity	10
5	Evaluation	11
6	Project Management	12
6.1	Project Definition	12
6.2	Project Planning	12
6.3	Project Execution	12

6.4 Project Completion	12
7 Conclusion	13

1 Introduction

Vera

1.1 Motivation

Vera

1.2 Usage Context

Vera

1.3 Project Goal

Vera

2 State of the Art

Laura

2.1 Image Forensic

Laura

2.2 Light Vectors

Laura

3 Materials

Laura

The following sections describe the resources and tools required for the completion of the project.

Furthermore, the test images are presented in chapter 3.3.

3.1 Hardware

Laura

During the implementation phase, the application was run on two computers, which are described in the following two sections. Both computers needed to be able to deal with the software components described in section 3.2. An extract from your data sheet is shown in table 1 respectively table 2.

3.2 Software

Laura

In order to develop the *Interactive Lighting Detector Qt* was used (compare section 3.2.1). To take advantage of already existing functionalities the *OpenCV*-library, which is described in section 3.2.2, was taken advantage of.

3.2.1 QT

Laura

3.2.2 OpenCV

Laura

The *Open Source Computer Vision* (OpenCV) is an open source library for image- and video processing, which is among others available in the programming language

NAME?	Description
Processor	??
RAM	??
Graphic Card	??
Operating System	??

Table 1: Extract from the Data Sheet of the NAME?

Acer Aspire 5820TG	Description
Processor	Intel Core i3 CPU @ 2.40 GHz
RAM	4 GB
Graphic Card 1	AMD Mobilty Radeon HD 5000 Series
Graphic Card 2	Intel(R) HD Graphics
Operating System	Windows 10 Education 64 bit

Table 2: Extract from the Data Sheet of the Acer Aspire 5820TG Notebook.

C++. It has been introduced ten years ago and is developed by various programmers since then. This library offers the most common algorithms, as well as current developments in image processing [1].

Für dieses System ist vor allem das Modul `calib3d` [?] und das extra Modul `aruco` [?] verwendet. Das erste Modul `calib3d` bietet alle notwendigen Funktionen zur Erstellung, Verwendung und Weiterverarbeitung von intrinsischen und extrinsischen Kamerakalibrierungen an (vgl. Abschnitt ??). Während das Zweite alle benötigten Ressourcen und Funktionalitäten zum Tracking von *ArUco* Markern zur Verfügung stellt (vgl. Abschnitt ??).

3.3 Testimages

Laura

3.3.1 First Batch

Laura



Figure 1: Bildunterschrift.

3.3.2 Second Batch

Laura

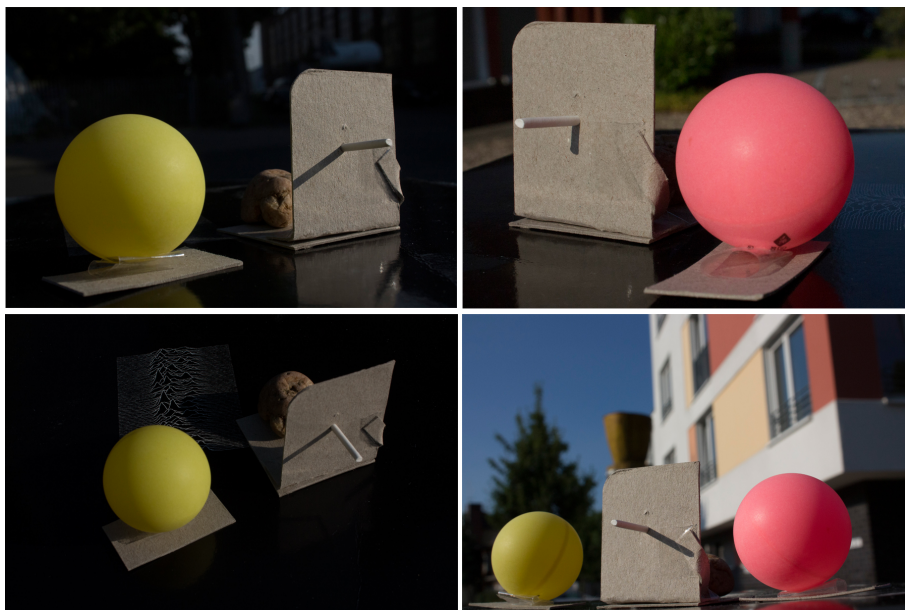


Figure 2: Bildunterschrift.

4 System

Vera

4.1 Lighting Model

Vera

4.2 Contours

Vera

4.2.1 Find Contours

Vera

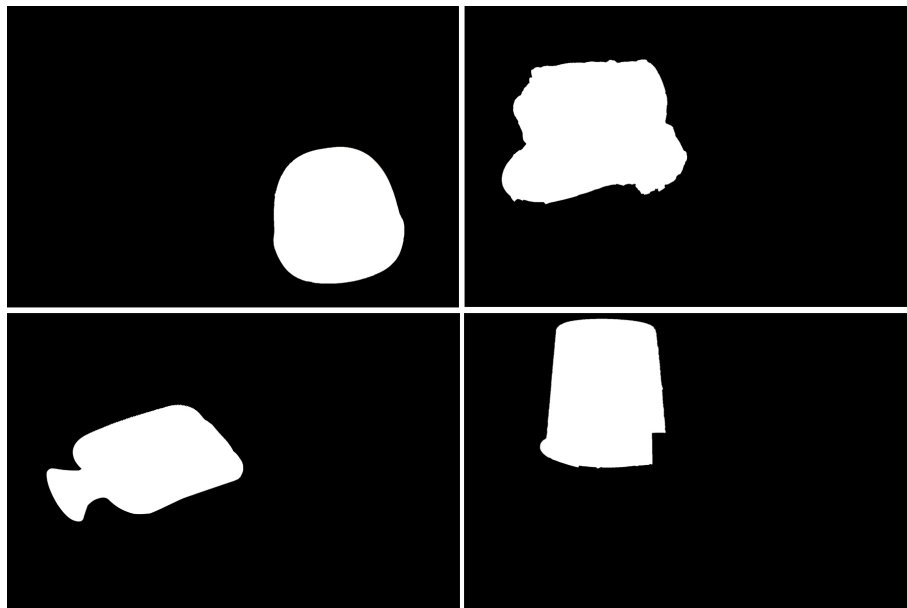


Figure 3: Bildunterschrift.

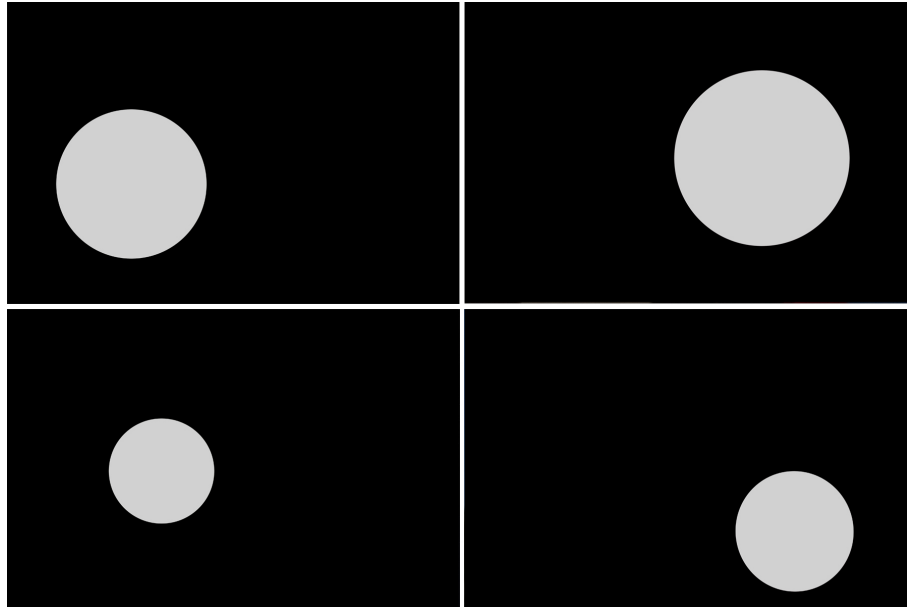


Figure 4: Bildunterschrift.

4.3 Subcontours

Vera

4.4 Different Approaches

Laura

4.4.1 1. Approach: One Lightvector

Laura

4.4.2 2. Approach: Averaging Lightvectors

Laura

4.4.3 3. Approach: Lightvector with highest Intensity

Laura

5 Evaluation

Vera und Laura: Stichpunkte

Vera: Ausformulierung

6 Project Management

Laura

6.1 Project Definition

Laura

6.2 Project Planning

Laura

6.3 Project Execution

Laura

6.4 Project Completion

Laura

7 Conclusion

Vera und Laura

References

- [1] I. Culjak, D. Abram, T. Pribanic, H. Dzapov, and M. Cifrek. A brief introduction to opencv. pages 1725–1730, May 2012.