AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE
WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ

KATEDRA AUTOMATYKI I INŻYNIERII BIOMEDYCZNEJ

Praca dyplomowa magisterska

Rozpoznawanie obiektów na obrazach RGB-D pod kątem zastosowań w robotyce

Object recognition in RGB-D images for robotic applications.

Autor: *Jakub Olesiński*
Kierunek studiów: *Automatyka i Robotyka*
Opiekun pracy: *dr inż. Paweł Rotter*

Kraków, 2016

Oświadczam, świadomy(-a) odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.

AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

FACULTY OF ELECTRICAL ENGINEERING, AUTOMATICS, COMPUTER SCIENCE AND BIOMEDICAL ENGINEERING

DEPARTMENT OF AUTOMATICS AND BIOENGINEERING

Master of Engineering Thesis

Object recognition in RGB-D images for robotic applications.

Author:	*Jakub Olesiński*
Degree programme:	*Automatics and Robotics*
Supervisor:	*dr inż. Paweł Rotter*

Kraków, 2016

Many thanks to all the people and beware global warming

Contents

Introduction

The aim of this work is to survey modern RGB-D image processing algorithms for model-based object recognition. Analysis of each method is focused on their usability in time and resource constrained robotic environment. Based on provided performance tests, selected methods will be used to develop a complete, applicable in robotics,object recognition system.

The first chapter provides project background. RGB-D imaging basics are introduced and followed by software tools and testing environment used throughout the project. Subsequently, scientific context of the work is settled.

Second chapter introduces preprocessing techniques. Firstly, different data representations and conversion methods are discussed. Afterwards, basic operations are introduced, including spatial transformations, neighbourhood calculation and normal estimation. Efficient noise filtering, data segmentation and keypoint extraction methods finalize this chapter.

Model fitting algorithms are presented in chapter three. Performance of data alignment and correspondence matching methods is tested together with different descriptor approaches. Statistical methods of result verification are finally presented and used to test the complete object recognition system.

Chapter four generalizes models from previously developed approach into categories. A neural network approach is presented as a data classifier. Convolutional neural network architecture is proposed and its implementation, training and performance tests are provided.

The last chapter describes several application scenarios of the developed system. Here should be placed one sentence per scenario. Here should be placed one sentence per scenario. Here should be placed one sentence per scenario. Each tested against different noise conditions.

1. Background

1.1. Motivation

Read and merge few article intros

The field of RGB-D image processing was developing slowly, until the release of Microsoft's Kinect in 2010. This low-cost camera gained significant scientific interest because of the potential it brought into the areas of virtual reality.

1.2. Software tools

Throughout this work, all of the software was developed with an extensive use of the Point Cloud Library (PCL)[1], which is an open source C++ library for 2D/3D image and point cloud processing. PCL provides implementations of a multitude of novel algorithms for 3D filtering, feature estimation, segmentation, registration and model fitting, together with tools for visualization and camera interfacing. Since 2011, it is developed by a large scientific community and maintained by the Open Perception foundation. PCL is an 3D equivalent of the OpenCV, the largest computer vision software library [ref]. Methods developed under this project were also aided by OpenCV tools, in places where classic 2D image processing was sufficient. Most promising algorithms analysed in this work was parallelized with CUDA GPU computing framework to optimize processing time.

1.3. Test environment

Testing of implemented algorithms was done on two hardware platforms: an embedded computing board NVidia Jetson TK1 and personal laptop Lenovo Y50-70. Parameters of both computers are summarized in table [x].

	NVidia Jetson TK1	Lenovo Y50-70
CPU:	ARM Cortex-A15 2.32GHz x4	Intel Core i7-4720HQ 2.60GHz x4
GPU:	NVIDIA Kepler GK20a	NVidia GeForce GTX 960M
	with 192 SM3.2 CUDA cores	with 640 SM5 CUDA Maxwell cores
	(upto 326 GFLOPS)	(upto xx GFLOPS)
RAM:	2GB DDR3L 933MHz 64 bit	16GB DDR3L 1600MHz (4GB GPU 2.5Mhz 128bit)

Table 1.1: Test hardware specification

Recognition system was validated against sample dataset from XXX, which provides full object models together with real world scene scans and ground truth position annotations. To

Print stanford bunny ?

Washington V2 Database - how to read it into pcl

YCB Benchmarks

1.4. Related work

2. Preprocessing

rgb-d, point clouds, tsdf algos for conversion, timing

2.1. Representation

segmentation

2.2. Neighbourhood

segmentation

2.3. Transformations

segmentation

2.4. Noise filtering

segmentation

2.5. Segmentation

segmentation

2.6. Keypoints

segmentation

3. Recognition

keypoints

3.1. Alignment

alignment

3.2. Matching

matching

3.3. Descriptors

descriptors

3.4. Verification

verification

4. Classification

4.1. Neural network

convolutional neural network

4.2. Training

training

4.3. Results

descriptors

5. Applications

5.1. Scenario 1

something simple with recognition
simulation of something simple

5.2. Scenario 2

something simple with classification
simulation of something simple

5.3. Scenario 3

Robocup@Work
simulation of Robocup@Work?

5.4. Scenario 4

APC?
simulation of APC?

Summary

Bibliography

[1] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, (Shanghai, China), May 9-13 2011.