Boosting for Generic 2D/3D Object Recognition

Dissertation

zur Erlangung des akademischen Grades
doctor rerum naturalium (Dr. rer. nat.)

vorgelegt dem Rat der Fakultät für Mathematik und Informatik
der Friedrich-Schiller-Universität Jena

von M. Sc. Doaa Abd Al-Kareem Mohammed Hegazy

geboren am 21. November 1979 in Kairo

Gutachter:

1. Prof. Dr.-Ing. Joachim Denzler, Friedrich-Schiller-Universität Jena
2. Prof. Dr. Daniel Cremers, Rheinische Friedrich-Wilhelms Universität Bonn

Tag der öffentlichen Verteidigung: 16. Dezember 2009

Ehrenwörtliche Erklärung

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet. Bei der Auswahl und Auswertung des Materials hat mich Prof. Dr.-Ing. Joachim Denzler unterstützt. Die Hilfe eines Promotionsberaters wurde nicht in Anspruch genommen und Dritte haben weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen. Die Dissertation wurde nicht in gleicher oder ähnlicher Form als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht. Auch habe ich noch keine andere Abhandlung bei einer anderen Hochschule als Dissertation eingereicht. Die geltende Promotionsordnung der Fakultät ist mir bekannt.

Doaa Abd Al-Kareem Mohammed Hegazy

Acknowledgments

First of all, I would like to thank Prof. Joachim Denzler for having welcomed me to join his research group, for supervising my work all the time, for his support and understanding and also for his patience.

I would like also to thank all the group members and all my colleagues for their continuous help and also for making me feel at home.

I would like to give special thanks to my husband for his continuous support all the time during the work on this thesis. Also, I would like to thank my daughter for giving me some of her beloved toys to build my datasets.

Abstract

Generic object recognition is an important function of the human visual system. Humans are able to categories different object classes in the surrounding environment in an easy way. For an artificial vision system to be able to emulate the human perception abilities, it should also be able to perform generic object recognition. However, for the machine, it is really a hard, complex and challenging task.

In this thesis, we address the generic object recognition problem and present different approaches and models which tackle different aspects of this difficult problem.

First, we present a model for generic 2D object recognition from complex 2D images. The model exploits only appearance-based information, in the form of a combination of texture and color cues, for binary classification of 2D object classes. Learning is accomplished in a weakly supervised manner using Boosting. The experimental results of the model are comparable, or outperform other state-of-the-art approaches.

However, we live in a 3D world and the ability to recognize 3D objects is very important for any vision system. Therefore, we present a model for generic recognition of 3D objects from range images. The problem of 3D object recognition is originally a hard task, and it is getting to be harder when recognizing 3D object classes is aimed. Our model makes use of a combination of simple local shape descriptors extracted from range images for recognizing 3D object categories, as shape is an important information provided by range images. Moreover, we present a novel dataset for generic object recognition that provides 2D and range images about different object classes. This dataset is considered to be the first to provide range images for different object categories. The range images of the dataset are acquired with a Time-of-Flight (TOF) camera. The dataset is used to build and evaluate our recognition model and promising classification results are obtained.

As the surrounding world contains thousands of different object categories, recognizing many different object classes is important as well. Therefore, we extend our generic 3D object recognition model to deal with the multi-class learning and recognition task. The learning process is adapted to allow the recognition of different object categories from range images. The model reveals good categorization performance despite the difficulty of the problem.

Moreover, we extend the multi-class recognition model by investigating the use of different information cues extracted from different data types for improving the categorization performance. A novel model which uses a combination of appearance-based information extracted from 2D images and range-based (shape) information extracted from range images is introduced for multi-class generic 3D object recognition and promising results are obtained. Generally, addressing the problem of generic 3D object recognition from range images is one of the main contributions of the work presented in this thesis.

Übersicht

Generische Objekterkennung ist eine wichtige Funktion des menschlichen Sehsystems. Für einen Menschen ist es ein Einfaches verschiedene Objekte in seiner Umgebung zu kategorisieren. Damit ein künstliches Sehsystem in der Lage ist, die menschlichen Sehfähigkeiten nachzuahmen, sollte es auch die generische Objekterkennung beherrschen. Allerdings stellt dies für eine Maschine eine sehr schwierige, komplexe und herausfordernde Aufgabe dar.

In dieser Arbeit wird das Problem der generischen Objekterkennung behandelt. Es werden mehrere Ansätze und Modelle zur Lösung verschiedener Aspekte dieses schwierigen Problems präsentiert. Zuerst wird ein Modell zur generischen 2-D-Objekterkennung von komplexen 2-D-Bildern vorgestellt. Dieses Modell verwendet zur binären Klassifizierung von 2-D-Objektklassen ausschließlich erscheinungsbasierte Information in Form von kombinierten Textur- und Farbmerkmalen. Das Lernen dieser Merkmale erfolgt unter geringer Beaufsichtigung (weakly semi supervised) mittels Boosting. In verschiedenen Experimenten zeigt sich, dass dieses Modell anderen Verfahren überlegen ist.

Da wir in einer dreidimensionalen Welt leben, ist die Erkennung von 3-D-Objekten sehr wichtig für jedes Sehsystem. Aus diesem Grund wird ein Modell zur generischen Erkennung von 3-D-Objekten unter Verwendung von Tiefenbildern vorgestellt. 3-D-Objekterkennung ist ein schwieriges Problem. Noch schwieriger ist jedoch die Erkennung von 3-D-Objektklassen. Die Objektform ist eine wichtige Information, welche in den Tiefenbildern enthalten ist. Um 3-D-Objektkategorien zu erkennen, verwendet das hier vorgestellte Modell eine Kombination aus einfachen, lokalen Formdeskriptoren, die aus den Tiefenbildern extrahiert werden.

Außerdem wird ein neuartiger Datensatz für generische Objekterkennung präsentiert, welcher Tiefen- und 2-D-Bilder von verschiedenen Objektklassen enthält. Dieser Datensatz ist der erste, der Tiefenbilder für verschiedene Objektklassen bereitstellt. Die Tiefenbilder werden mit einer Time-of-Flight-Kamera (TOF) aufgenommen. Der Datensatz wird zur Auswertung unseres Erkennungsmodells verwendet und es werden überzeugende Ergebnisse erreicht.

Da die uns umgebende Welt aus tausenden verschiedenen Objektkategorien besteht, ist das Erkennen vieler verschiedener Objektkategorien ein weiteres wichtiges Problem. Aus diesem Grund wird eine Erweiterung des 3-D-Objekterkennungsmodells vorgestellt, die für Mehrklassenobjekterkennung geeignet ist. Der Lernprozess wird so angepasst, dass er die Erkennung von verschiedenen Objektkategorien auf Basis von Tiefenbildern ermöglicht. Trotz der Schwierigkeit des Problem erreicht das vorgestellte Verfahren sehr gute Erkennungsraten.

Des Weiteren wird unser Mehrklassenerkennungsmodell so erweitert, dass es in der Lage ist, Merkmale verschiedener Datentypen zu verwenden, um die Klassifizierungsleistung zu verbessern. Dazu wird ein neuartiges Modell zur 3-D-Objekterkennung vorgestellt, welches eine Kombination aus erscheinungsbasierter Information aus 2-D-Bildern und formbasierter Information aus Tiefenbildern verwendet. In den Experimenten erreicht dieses Modell vielversprechende Ergebnisse.

Die Behandlung des Problems der generischen 3-D-Objekterkennung auf Basis von Tiefenbildern ist einer der Hauptbeiträge dieser Arbeit.

Contents

List of Figures **xiii**

List of Tables **xvii**

1 Introduction **1**

1.1 Motivation 1

1.2 Generic Object Recognition 2

1.2.1 Challenges 3

1.3 Contributions 8

1.3.1 Generic 2D Object Recognition Model 9

1.3.2 New 2D/3D Object Category Dataset 9

1.3.3 Generic 3D Object Recognition Model 10

1.3.4 Combination of Appearance and Range-based Information for Generic 3D Object Recognition 10

1.4 Thesis Outline 10

2 Theoretical Preliminaries and Important Issues **13**

2.1 Representation of Objects 13

2.1.1 Global Representation 14

2.1.2 Local Representation 14

2.2 Learning Objects Representations 18

2.2.1 Learning with Boosting 19

2.3 Classification 27

2.4 GOR Datasets 28

2.4.1 2D Images Datasets 29

2.4.2 A Dataset for 3D Object Categories 32

2.4.3 Range Images Datasets 33
2.5 Evaluation of Recognition 34
2.6 Conclusions 37

3 Related Work **41**
3.1 Specific Object Recognition 42
3.1.1 Geometrical and Model-based Approaches 42
3.1.2 Appearance-based Approaches 44
3.2 Generic Object Recognition 47
3.2.1 Appearance-Based Approaches 47
3.2.2 Shape-Based Approaches 57
3.2.3 3D Recognition Approaches 57
3.2.4 Multi-Class Recognition Approaches 60
3.3 Conclusions 61

4 Generic Recognition of 2D Objects **63**
4.1 Motivation 63
4.2 The Recognition Model 64
4.2.1 Overview 64
4.2.2 Interest Points Detection 66
4.2.3 Regions Description 67
4.2.4 Learning Model 68
4.3 Experimental Evaluation 70
4.3.1 Experiments using Caltech 4 dataset 70
4.3.2 Experiments using Graz02 Dataset 79
4.4 GLOH vs. SIFT: Performance Evaluation 82
4.4.1 Using Caltech 4 Dataset 83
4.4.2 Using Graz02 Dataset 83
4.5 Evaluation of Boosting Algorithms on GOR 89
4.5.1 Used Weak Learner 89
4.5.2 Experiments and Results 90
4.6 Conclusions 93

5 A 2D/3D Object Category Dataset **95**
5.1 Motivation 95
5.2 3D Image Acquisition Methods 96
5.2.1 Time-of-Flight PMD Camera 98

5.3 JenaRange01 2D/3D Dataset 100
5.3.1 Dataset Acquisition Process 101
5.4 JenaRange02 2D/3D Dataset 104
5.4.1 Dataset Acquisition Process 105
5.5 Experimental Evaluation 109
5.6 Conclusions 112

6 A Model for Generic 3D Object Recognition 113
6.1 Motivation 113
6.2 The Recognition Model 115
6.2.1 Overview 115
6.2.2 Preprocessing and Interest Regions Detection 116
6.2.3 Local Features Computation 116
6.2.4 Learning Model 120
6.3 Experimental Evaluations and Results 120
6.3.1 Experiment 1: Generic Recognition Performance 121
6.3.2 Experiment 2: Recognition Performance with Respect to Viewpoint Change 122
6.3.3 Experiment 3: Generic Recognition Performance with Respect to Viewpoint Change 123
6.3.4 Experiment 4: Generic Recognition in Scenes with Multiple Objects 124
6.4 Conclusions 128

7 Multi-Class Recognition 129
7.1 Motivation 129
7.2 Multi-Class Recognition Model 130
7.2.1 Model Overview 130
7.2.2 Learning Model 130
7.3 Experimental Evaluation 131
7.3.1 Experimental setup 131
7.3.2 Results 133
7.4 Dense Grid Sampling of Range Images 138
7.4.1 Experiments and Results 139
7.5 Conclusions 143

8 Fusion of Appearance and Range Information 147
8.1 Motivations 147
8.2 Model Overview 148
8.2.1 Local Description 149
8.2.2 Learning 150
8.3 Performance Evaluations 150
8.3.1 Settings 150
8.3.2 Using Point Detection for Sampling Range Images 151
8.3.3 Using Regular Grid for Sampling Range Images 156
8.4 Conclusions 161

9 Summary, Conclusions and Future work 163
9.1 Work Summary 163
9.2 Contributions and Outcomes 164
9.2.1 Summarization of Best Results 165
9.3 Conclusions 168
9.4 Future work 170

Bibliography 171

List of Figures

1.1 Specific and generic object recognition. 3
1.2 General challenges of generic object recognition 5
1.3 Functional and visual categories 6
1.4 Intra- and inter-class variabilities 7
1.5 3D object recognition 8

2.1 Example of local representations of 2D images 17
2.2 Hard vs. Soft margins 26
2.3 Example images of the different object classes of ETH-80 dataset. . . 30
2.4 Example images of the different object classes of Caltech 4 dataset. . 31
2.5 Example images of the different object classes of UIUC dataset. . . . 31
2.6 Example images of the different object classes of Graz01 dataset. . . 32
2.7 Example images of the different object classes of Graz02 dataset. . . 33
2.8 Example images of the different object classes of the 3D Object Categories dataset of Savarese and Fei-Fei in [130]. 34
2.9 Example images of OSU/SAMPL range dataset. 35
2.10 Example images of Stuttgart range dataset. 36
2.11 ROC and RPC curves 38

3.1 Shape context computation and matching [22]. 44
3.2 The SCERPO vision system 45
3.3 The codebook representation [13] 48
3.4 Stages of the model of Fritz *et al.* [52] 49
3.5 Differences between standard codebook approaches and the approach of Yang *et al.* [160] 50
3.6 A face model (with 6 parts) developed by [43] 52

3.7 Recognition model of Viola and Johns [151] 53
3.8 An overview of the recognition model of Zahng *et al.* [162]. 54
3.9 The classification model of Moosmann *et al.* [105] using ERCF and adaptive saliency map. 55
3.10 The biologically-based GOR model of Mutch and Lowe [108] 56
3.11 An example of input and matching results using the Pictorial structure of Felzenszwalb and Huttenlocher [42] 58
3.12 The main idea of the model of Savarese and Fei-Fei [130] 59
3.13 Recognition model for shape classes from range images proposed by Ruiz-Correa *et al.* in [129] 60

4.1 The general framework of our model for generic 2D object recognition. 65
4.2 An example describing the different phases of our recognition model. 66
4.3 SIFT descriptors 67
4.4 Example images of the two classes used as counter-class in the Caltech 4 experiments 73
4.5 Performance comparison of the performance of the recognition model using GLOH-color combination (GC) on the Caltech 4 dataset 76
4.6 Example of incorrectly classified images on the Caltech dataset using the GLOH-color combination (GC-G). 77
4.7 ROC curves of the performance of the recognition model using the GLOH and color combination (GC) on the Graz02 dataset. 81
4.8 Example of incorrectly classified images from the Graz02 dataset using the GLOH-color combination (GC). 82
4.9 Comparison between the performance of the recognition model using GC combination and the SC combination on the Caltech4 dataset using the Graz01 background class as a counter-class. 85
4.10 Comparison between the performance of the recognition model using GC combination and the SC combination on the Graz02 dataset. . . . 88

5.1 Illustration of the principle of the Time-of-Flight rangers. 98
5.2 Functional block diagram of the image processing chain of (a) Stereo vision systems and (b) TOF vision systems [66]. 99
5.3 Phase measurements by averaging mixed reflected and sent-out waveforms [109]. 100
5.4 The three visual classes of the new 3D object category database JenaRange01 102

5.5 The acquisition procedure of the dataset images. 102
5.6 Example 2D images of the dataset for the three visual classes 103
5.7 Distribution of range means of the dataset images for the three object classes. 104
5.8 Example images of the five object classes of the new object category dataset JenaRange02. 106
5.9 Example images of the JenaRange02 dataset at eight different viewpoints as well as the four different heights. 107
5.10 Example images of the JenaRange02 dataset with occlusion and truncation. 108
5.11 Example range images (using TOF PMD camera) and their corresponding color images (using CCD camera) of the JenaRange02 dataset . . 108
5.12 Object class instances used to train and test the recognition model. . . 110
5.13 Object class instances used to train and test the recognition model. . . 111

6.1 Example images of the dataset used in [129] and our dataset JenaRange01 115
6.2 The proposed generic 3D object recognition model. 117
6.3 The process of interest points detection in the used range images. . . 118
6.4 Representation of normals in sphere coordinates [63] 119
6.5 Instances of the object classes of the JenaRange01 dataset used for training and testing the recognition model 121
6.6 The ROC curves of the three classes on the generic recognition task (experiment 1). 123
6.7 Example training and test images for experiments 2 124
6.8 Example test images used in experiment 3. 125
6.9 Example of the images recorded for the task of recognition in scenes with multiple objects. 126
6.10 The ROC curves of the three classes on the categorization with the presence of clutter and occlusion task. 127
6.11 Examples of incorrectly classified images (false negatives) of experiment 4 127

7.1 Instances of the different object classes used to train and test the recognition model. 134
7.2 Examples of incorrectly classified images from JenaRange02 dataset using range-based information 136
7.3 Example of range images description using dense grid. 139

7.4 Dense grid sampling for range images 140
7.5 Effect of the parameters N and M on the recognition performance of the model 141
7.6 Comparison of multi-classification results between the range images sampled using points detection (Point) and dense grid (Grid) respectively 145

8.1 The general framework of the proposed GOR model based on different information fusion. 149
8.2 Performance comparisons of categorization performance using range-based (Shape), appearance-based (SIFT-color) and appearance-range based (Shape-SIFT-color) combination respectively 153
8.3 Examples of incorrectly classified images from JenaRange02 dataset using appearance and range-based information fusion 158
8.4 Performance comparisons of categorization performance using range-based (Shape), appearance-based (SIFT-color) and appearance-range based (Shape-SIFT-color) combination respectively 159

List of Tables

4.1 Classification rates at ROC-eqq.-err. rates of our recognition results using the Caltech 4 dataset with different counter-classes (Caltech 4 background, Graz01 background and Leaves class). 74

4.2 Classification rates at ROC-eqq.-err. rates of our results using (GC) feature combination on the Caltech 4 dataset with using the Graz01 background class as the counter-class. 75

4.3 Classification rates at ROC-eqq.-err. rates of our results using (GC) feature combination on the Caltech 4 dataset with using the leaves class as the counter-class. 75

4.4 Classification rates at ROC-eqq.-err. rates of our recognition model using the Caltech 4 dataset compared to state-of-the-art approaches. . 78

4.5 Classification rates at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using GLOH, color and GLOH-color combination (GC) respectively. 80

4.6 Classification rates at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using the GC combination compared to the state-of-the-art approaches. 82

4.7 Classification rates at ROC-eqq.-err. rates of our recognition model using SIFT-color combination (SC-G) on the Caltech 4 dataset with the Graz01 background class as a counter-class. 84

4.8 Classification rates at ROC-eqq.-err. rates of our recognition model using SIFT-color combination (SC-L) on the Caltech 4 dataset with the leaves class as a counter class. 84

4.9 Comparison between the performance of the recognition model using (GC-G) combination and the (SC-G) combination on the Caltech 4 dataset compared to state-of-the-art approaches. 86

4.10 Classification rates at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using SIFT, color and SIFT-color combination (SC) respectively. 87
4.11 Classification rats at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using the SC combination compared to the state-of-the-art approaches. 87
4.12 Generalization performance of Discrete AdaBoost, Real AdaBoost and SoftBoost algorithms represented by classification rates ROC-eqq.-err. rates. 92
4.13 Generalization rates (classification rates at ROC-eqq.-err. rates) of Discrete AdaBoost, Real AdaBoost and SoftBoost algorithms using training data with different degrees of label noise. 93

5.1 Classification rates at ROC-eqq.-err. rates of recognition using JenaRange02 compared to the results using Graz02. 110
5.2 Classification rates at ROC-eqq.-err. rates of recognition using JenaRange02 compared to the results using Graz02. Different counter-classes are used. 111

6.1 ROC-eqq.-err. rates of the generic performance of the used three object classes (Experiment 1). 122
6.2 ROC-eqq.-err. rates of the generic recognition performance of the used three object classes with respect to viewpoint change (experiment 3. . 125
6.3 ROC-eqq.-err. rates of recognition using complex scenes for the used three object classes. 126

7.1 Classification performance at the ROC-Equal-Error rates of the multi-class recognition using range-based information 135
7.2 Confusion matrices of multi-class recognition using range-based information 137
7.3 Confusion matrices of multi-class recognition using range-based information with dense grid sampling of range images 142
7.4 Classification performance at the ROC-Equal-Error rates of the multi-class recognition using grid representation of range images 143

8.1 Classification performance at the ROC-eqq.-err. rates of the multi-class recognition using SIFT-color features (SC) combination 152

8.2 Classification performance at the ROC-eqq.-err. rates of the multi-class recognition using Shape-SIFT-color features combination 152
8.3 Confusion matrices of multi-class recognition using SIFT-color features (SC) combination 154
8.4 Confusion matrices of multi-class recognition using Shape-SIFT-color features combination 155
8.5 Classification performance at the ROC-eqq.-err. rates of the multi-class recognition using Shape-SIFT-color features combination 158
8.6 Confusion matrices of multi-class recognition using Shape-SIFT-color features combination 160

9.1 Best results achieved on Caltech 4 dataset. 166
9.2 Best results achieved on Graz02 dataset. 166
9.3 Best results achieved on JenaRange01 dataset dataset. 167
9.4 Best results achieved on JenaRange02 dataset dataset for multi-class recognition using range-based information. 168
9.5 Best results achieved on JenaRange02 dataset dataset for multi-class recognition using appearance and range-based information fusion. . . 168

Chapter 1

Introduction

This chapter provides a definition of the *generic object recognition* (GOR) problem and shows why it is important for the computer vision community to find a solution to this problem. Then , it sheds light on the difficulties of the problem and explains why finding an appropriate solution for it is a hard and challenging task. Afterwards, the contributions of the work presented and mentioned in this thesis are briefly stated. An overview of the outline and organization of the thesis is finally given.

1.1 Motivation

Developing machine vision approaches that emulate some of the *human visual system* (HVS) abilities is the goal of almost all computer vision researchers. The ability to recognize objects in the surrounding environment is one of the most important abilities of HVS and, for long time, researchers are trying to develop machine vision models that can get closer to this ability. Many models and system are built that are able to recognize many different previously seen objects under different environmental conditions such as size and illumination variations with high accuracy. This type of recognition is called *specific object recognition*. The specific object recognition task is been solved for almost all of the cases. However, for a recognition model to be able to emulate the human recognition abilities in a right way, it should be able to recognize thousand of different objects in the environment. It is hard and difficult to fulfill this goal using specific object recognition because the recognition model should in this case learn all these different objects, each separately, which is a time, resources and effort consuming process.

By observing the HVS, it can be noticed that humans are able to recognize objects which are visually similar and classify them to the same object class. They are able to recognize any previously unseen object which share some visual similarities with another known (previously learned) object to belong to the same class. Therefore, we can say that humans learn and recognize objects in generic groups rather than separately. Figure 1.1 gives an example to both specific and generic object recognition. Instead of learning all previously unseen specific objects separately, human put them in categories to ease his life. For this reason, generic object recognition is essential to understand images and scenes.

Unfortunately, machines do not have the algorithm to perform generic object recognition like humans or even to approach their ability in performing this task. Therefore, this is one key problem in computer vision

Generic object recognition is important for many different human daily as well as industrial applications such as video, web, and databases search, security, robotics, navigation and many other applications.

The outline of this chapter is as follows. Section 1.2 provides a precise definition of the generic object recognition problem and explains its difficulties. It briefly discusses the challenges faced by the researchers when they try to tackle the problem and find a solution to it. Section 1.3 briefly shows the contributions of the work presented in this thesis. Finally, an outline of the thesis's organization is given in section 1.4.

1.2 Generic Object Recognition

We can define the generic object recognition task as the process of assigning a specific object to a certain category [118]. Generic object recognition is also termed as ”visual class recognition” or ”object categorization”. As we previously mentioned, generic object recognition is different from specific object recognition, as the second considers the recognition of specific, individual objects. Object categories are those exist in the world around us like for example cars, bikes motorbikes, children or animals while specific objects can be my child, my bike or the sky tower (see figure 1.1).

Comparing the recognition abilities and performance of the humans with artificial recognition systems or models, it turns out that humans are much better in categorization than machines, but specific object recognition can often be handled more efficiently, reliably or simply faster by an artificial recognition model [118].

Although generic object recognition is important, there is no practical solution for it yet. This is due to the challenges faced by the researchers when they handle the prob-

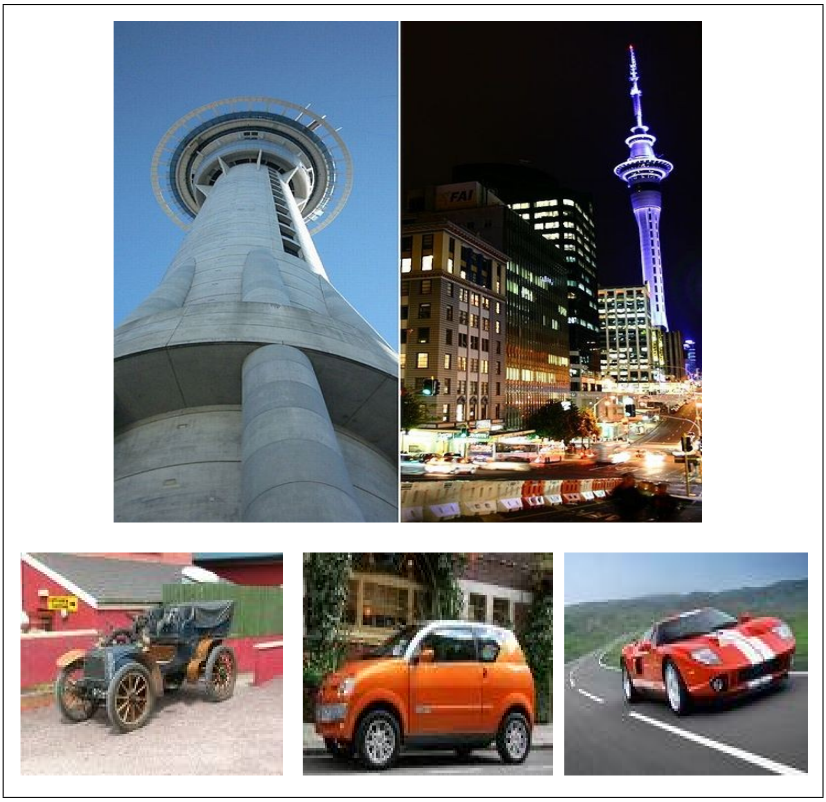


Figure 1.1: *Specific and generic object recognition.* In the first raw, an example of the specific object ”sky tower” is shown. The second raw gives an example of three instances of the visual class ”cars”.

lem. Following, we mention these challenges and their effect on appropriately solving the problem.

1.2.1 Challenges

The generic object recognition is still a challenging and unsolved task as we previously mentioned. It inherits the difficulties of the specific object recognition problem in addition to its own difficulties. There are many research done in this area but most of them did not focus on all the different aspects of the problem at once. They focused on some and simplified the others to suit their purpose. Here, we provide a short description of the challenges that must be faced for solving the problem.

General Challenges

- **Size and viewpoint variations**: The size of objects as well as their viewpoints can change from an image to another as shown in figure 1.2 (a). Therefore, any recognition model should be able to cope with these size and viewpoint variations.
- **Illumination variations**: The lighting in the images can change causing changes in the values of images pixels. This change can be a shift or scaling in the pixel values. An example for illumination variation is given in Figure 1.2(b). Robust recognition despite illumination variations is required.
- **Background clutter**: Objects are rarely found alone in the world. They are always exist with many different objects, which form then the background of the images as shown in figure 1.2 (c). The existence of background clutter imposes recognition difficulties of the required object.
- **Occlusion and truncation**: Objects in the images can be partially hidden or obscured by another objects (occlusion) or by part of the object itself (partial occlusion). Moreover, part of the object can be missed by the image boundaries (truncation). See figure 1.2 (d) and (e) for examples. The ability of any recognition system to recognize objects despite occlusion and truncation is important.

Definition of Object Category

It is concerned with what do we mean by object categories and classes and on which bases objects can be grouped into categories. Actually, there are two ways to group objects into classes: visually and functionally. In the visual grouping, objects which are related by some kind of visual consistency or share the outline of the appearance are grouped together into categories. In contrast, a functional grouping is that objects which are used for the same function are grouped together. In this case, the objects might share the same visual appearance but they do not have to. Figure 1.3 gives an example of both different grouping methods.
However, modeling functional categories of objects can not be done using only visual

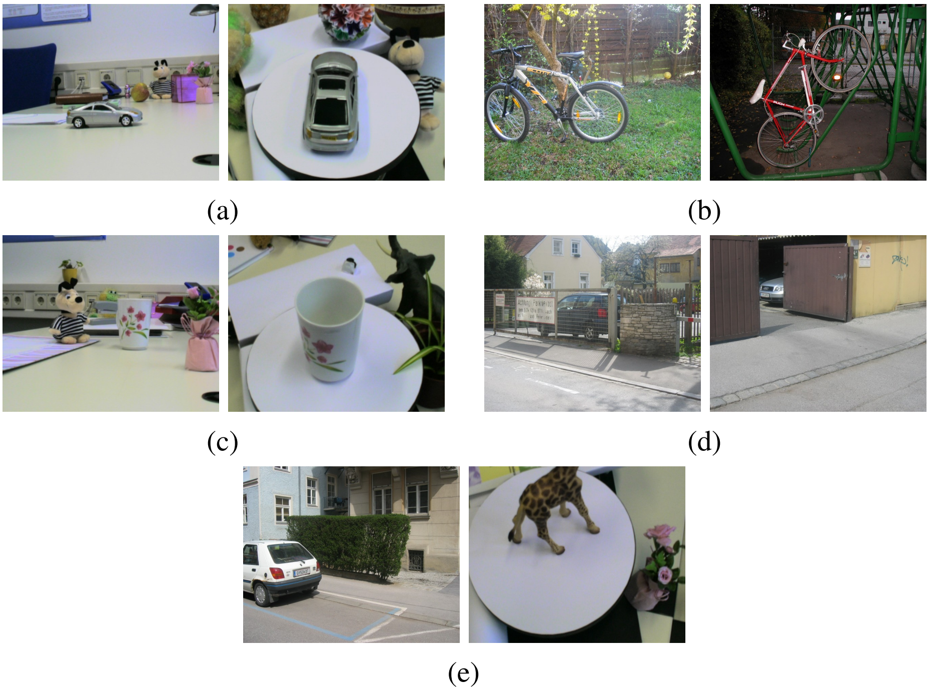
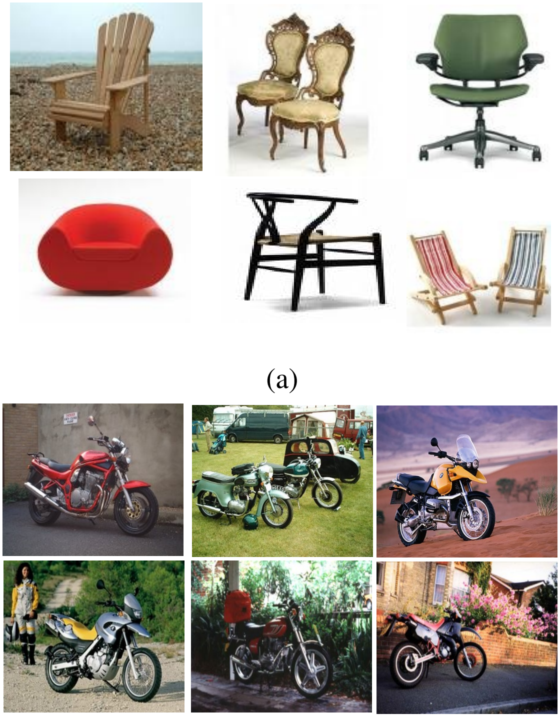

(a) (b) (c) (d) (e)

Figure 1.2: *General challenges of generic object recognition*. (a) Size and viewpoint changes. (b) Illumination variations. (c) Background clutter. (d) Occlusion. (e) Truncation.

appearance information. Therefore, most generic object recognition methods, including those are described in this thesis, limit and concentrate the recognition to the objects which are visually grouped.

Intra- and Inter-class Variabilities

The most challenging aspect of the generic object recognition problem is dealing with the intra- and inter-class variabilities within and among the different object classes. Within the same visual category, different degrees of appearance variations can exist among the objects as shown in figure 1.4 (first raw). Among the different visual categories, some appearance similarities can occur as shown in the example given in the



(a)

(b)

Figure 1.3: *Functional and visual categories*. (a) Example of functional grouping of objects forming the functional class ”chairs”. As shown, the visual appearance variations among the class instances is very wide. (b) Example of visual grouping of objects forming the visual class ”motorbikes”.

second raw in figure 1.4.

Therefore, any approach or model proposed for solving the problem should, on one hand, handle the appearance variations exist among objects that belong to the same visual class (intra-class variabilities). On the other hand, it should not confuse between objects of different object classes (inter-class variabilities).

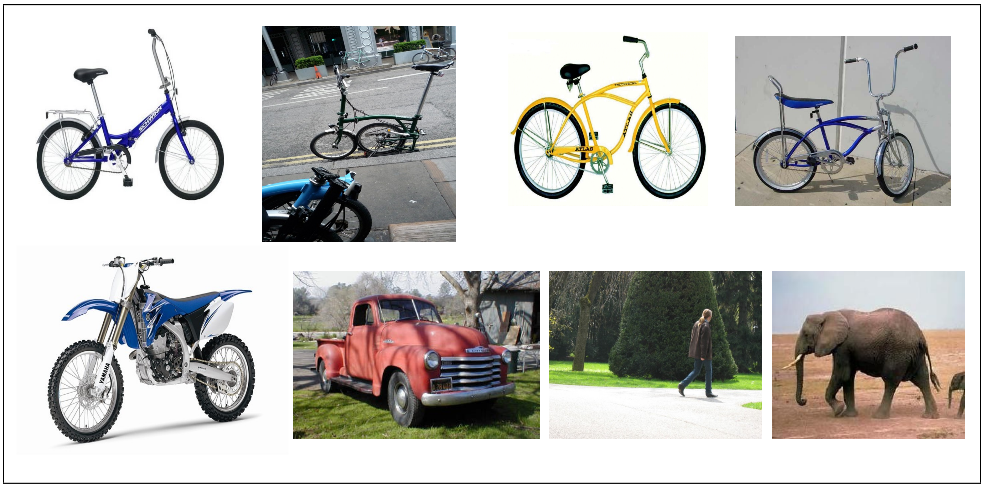


Figure 1.4: *Intra- and inter-class variabilities.* First raw gives examples to the intra-class variabilities within the visual class ”bikes”. Second raw shows inter-class variabilities examples among different visual classes and the class ”bikes”. Some of these classes such as the class ”motorbikes” (left most image) have some visual similarities to the class ”bikes”.

Amount of Supervision in Learning

The amount of supervision needed to train a model is an important and challenging issue in the problem. It is clear that with increasing the amount of supervision while training a recognition model, the recognition performance increases. Supervision in the form of object segmentation or bounding objects with boxes in the images is effort and time consuming. However, the weakly supervised learning, in which only labels are given to the images indicating their class type, is considerably enough.

Recognizing Many Categories

For a recognition model to be comparable to the human recognition abilities, it should be able to recognize many different object classes. However, generic object recognition is a difficult problem in general for only one class and it becomes more challenging when recognizing many different classes is aimed.

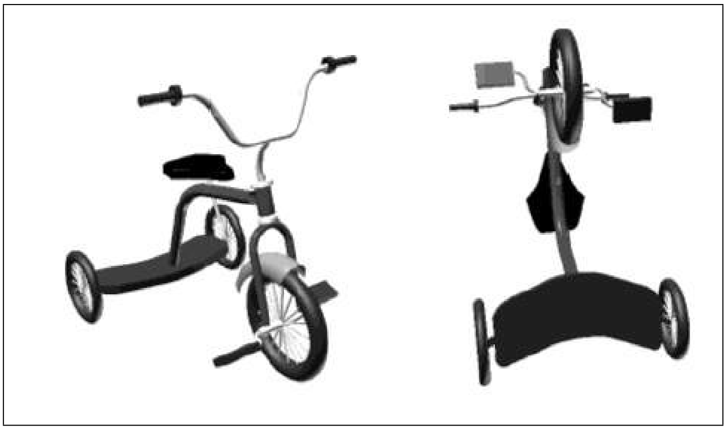


Figure 1.5: *3D object recognition.* Left image: a certain view of a 3D object (tricycle) that is easy to be recognized . Right image: the object is seen from an uncommon viewpoint, which impose some difficulties on the recognition process [117].

3D Object Recognition

When generic 3D object recognition is aimed, more challenges are added to the problem. 3D objects have infinite number of different viewpoints depending on where the viewer is standing and how the camera is oriented. Some of these viewpoints are uncommon and hard to be classified by even by humans such as the ”tricycle” example given in figure 1.5. The ability to handle and cope with this large viewpoint variations is then a mandatory requirement for a recognition model designed for accomplishing this recognition task.

1.3 Contributions

This thesis addresses the problem of 2D and 3D generic object recognition and presents four main contributions:

- A model for generic 2D object recognition using a combination of local appearance information.
- A new generic object recognition dataset which provides different types of data (2D and range images) of different object categories.

- A model for generic 3D object recognition using range images and its extension to the multi-class recognition case.
- A combinational model of appearance and range-based information for multi-class generic 3D object recognition.

The object recognition models learn the different object categories from training sets with weak supervision. The models differ in the type of data used for recognition. The first model, for generic 2D object recognition, makes use of only appearance information extracted from 2D images, while the second one, for generic 3D object recognition, exploits local shape cues computed from range images. The third one uses a combination of the two different information types. Boosting is the underlaying learning technique in all the different models. Additionally, we present a new object category dataset which provides 2D as well as 3D (range) images of different object class. Moreover, the 3D recognition model is extended to multi-class recognition. A final model which combines different feature types like color, grayscale, and shape, extracted from different types of images, is presented for improving the recognition performance for multi-class generic 3D object recognition model.

1.3.1 Generic 2D Object Recognition Model

Our first model is proposed for generic 2D object recognition. The model is an appearance-based one which uses a combination of two different appearance cues, represented in texture and color, for generic object recognition. No geometrical or spatial information is used for recognition and labels of the training images are the only information giving to the model during learning (weakly supervised learning). Performance evaluation of the model using different generic object recognition benchmarks is performed which revealed its good classification performance.

1.3.2 New 2D/3D Object Category Dataset

Up to our knowledge, there is no dataset for generic object recognition which provides range images of different object categories. Therefore, we constructed and presented a new 2D/3D generic object recognition dataset which provide two different image types of its members: 2D as well as range images. Two versions of the dataset are constructed, namely ”JenaRange01” and ”JenaRange02”. They differ in the number of object classes, viewing and image acquisition procedures and degree of difficulty.

Moreover, an experimental evaluation of the dataset and performance comparison to a popular benchmark reveals its difficult nature.

1.3.3 Generic 3D Object Recognition Model

With the aid of our new dataset, we developed an approach for generic 3D object recognition. The model depends on information extracted from range images in the form of simple local shape features. The framework of the proposed model is usually applied for recognition form 2D images and never been applied for recognition from range images.

Moreover, the model is extended for multi-class generic 3D object recognition, which is more difficult problem. It is worth saying, that our model is the first one (up to our knowledge) that address the difficult and challenging problem of generic 3D object recognition from range images.

1.3.4 Combination of Appearance and Range-based Information for Generic 3D Object Recognition

A combination of different information cues such as color, texture and shape, extracted from 2D and range images, is used to improve the performance of the multi-class 3D object recognition model. This combination is one of our contributions and never been presented before for generic 3D object recognition or even generic object recognition from range images.

1.4 Thesis Outline

The organization of this thesis goes as follows. Chapter 2 sheds light on the different and important aspects for generic object recognition such as objects representation, classification and methods for models evaluations. It gives a brief explanation to each aspect and mentions its importance for solving the problem. Moreover, it presents a brief overview of the Boosting learning technique, which is used for learning in all the recognition models presented in this thesis.

Chapter 3 gives an overview of the different models and approaches proposed and developed for giving a solution to the object recognition problem in general and the generic object recognition in specific. The overview covers the area of specific and generic 2D object recognition, giving a brief description to the state-of-the art approaches. Moreover, approaches developed for 3D object recognition from both 2D

and range images are also reported.
Afterwards, chapter 4 introduces our model for generic 2D object recognition. It gives an overview of the model, followed by detailed explanations of its different phases and ended by experimental evaluations of the model performance.
Chapter 5 describes our new object category dataset with its two different versions. First, it provides an overview of the different range imaging methods including the technique used in acquiring the images of our dataset. Then, it describes dataset with its two versions as well as the acquisition procedures.
Then chapter 6 moves us to the more difficult problem of generic 3D object recognition and presents our proposed model for solving such challenging problem using range images.
Chapter 7 shows how the model for generic 3D object recognition is extended to handle the multi-class recognition case, which is more realistic than dealing only with binary classification.
Chapter 8 shows how different information of objects such as color, texture and shape, extracted from different data types (2D and range) can be combined to increase the recognition performance.
Finally, chapter 9 summaries the contributions and outcomes presented in this thesis and draws conclusions on the obtained results. Additionally, it provides ideas of the possible further research directions.

Chapter 2

Theoretical Preliminaries and Important Issues

This chapter provides an overview of the main and important issues for generic object recognition (GOR) such as how object categories are represented, learned and classified. Moreover, an overview of the current available GOR datasets as well as the recognition evaluation methods is presented. Many different issues for GOR are discussed, showing how these issues should be put into consideration when designing a model for GOR.

2.1 Representation of Objects

How objects and visual classes are represented for recognition is important for building a GOR model. Normally, data of different visual classes is given in the form of a set of input events (e.g. images). In this case, objects or classes representation is the representation of these events with extracting valuable information from them. The representation of objects can cover many aspects of their properties such as color, texture and discontinuities (edges, corners and lines). Moreover, the representation can cover more complete aspects of objects or visual classes such as shape and geometry. Even simple spatial relations can be modeled from the given data (images coordinates). The observed world is a three dimensional (3D) space and information is lost when a scene is projected to 2D. Therefore, changing occurs to the captured features when certain parameters of the image capturing process are changed, such as geometrical or illumination variations. More examples of these variations are given in section 1.2.1.

Finding a representation which provides *invariance* to such variations is, therefore, of great importance in building GOR models and systems.

Following, we give an overview of the different image representation techniques. We briefly discuss the global and local image representation with emphasizing the important differences between them. Then, we concentrate on the local representation of images [1] and explain the different methods used for 2D and range images.

2.1.1 Global Representation

Many Object recognition models, especially the early ones, used global representations to describe images. Global representation (features) describe the image as a whole and produces a very compact representation of it. Global features have the advantages that the whole image is described using only a single vector, which is used in slandered classification techniques in a straightforward way. However, they are sensitive to occlusion and background clutter. Moreover, images which contain only a single object are assumed to be used or a good segmentation of objects from the background is assumed to be available.

Global features include shape descriptors such as shape index [76, 36], contour representation such as Moment invariants [65], and texture features such as the local binary patterns [110].

2.1.2 Local Representation

Instead of describing the whole image using only one feature vector as in the global representation, components, regions or patches of the images are described separately. These regions can be sampled on a regular grid, at random or at selected interest points that detected by applying some interest operator to the images. Afterwards, the sampled regions are extracted and described using a suitable descriptor. Images are then described by multiple feature vectors.

Local representation (features) of images has the advantages that they are robust to occlusion and clutter. However, specialized classification algorithms might be required to handle cases in which a variable number of feature vectors per image exist. Despite this disadvantage, local features have been used very successfully in the development of current GOR systems.

---

[1] Since local representation is used in all the recognition approaches described in this thesis.

Representation of 2D Images

Local Regions/Patches Sampling As we previously mentioned, the local regions or patches in the images can be sampled in different ways. Either on a regular grid [26], at random locations in the images [93] or at selected points detected by an interest point detector.

Local descriptors computed from interest regions have proved to be very successful in many applications such as object recognition [92, 114] and image retrieval [148]. They are distinctive, robust to occlusion, background clutter and image transformation and do not need segmentation. The idea is to detect and extract image regions covariant to a class transformation, which are used as support regions to compute invariant descriptors. Given these invariant regions, finding appropriate descriptor(s) to describe these regions, and which is at the same time suitable for the application, is the remaining task [101].

Interest point detectors use different image measurements and can be invariant to many different transformations. Most of the traditional detectors follow the same following procedure. First, a saliency map is computed, which is a local function of the image. The saliency is a measure of local information content in the image or the local image contrast. Patches with high contrast (corners or highly textured areas) are expected to be detected and localized reliably between different images of the scene. Therefore, the local maxima of the saliency map are selected as features. This process is repeated after sub-sampling the image iteratively, to provide a multi-scale detector. Only local maxima that exceeds a certain threshold is finally selected to provide some invariance to noise [106].

Example of detectors are Forstner detector [48], Harris detector [56], Hessian detector [21], Difference-of-Gaussian (DoG) detector [33], Kadir-Brady detector [74] and MSER detector [94]. Some detectors provide scale invariance such as Harris-Laplacian detector [100] and Hessian-Laplacian [102]. Moreover, some detectors provide affine invariant patches (regions) such as Harris and Hessian affine detectors [99, 102].

The performance of many detectors has been evaluated in many different contexts such as [102, 101] in the context of viewpoint invariant matching, [98] in object class recognition and [106] in 3D object recognition.

Local Descriptors After the interest points are detected in the given images, interest regions (of scale dependent size) are cropped out. These regions are, afterwards, characterized or described somehow using a local description method. There is a large number of possible descriptors which emphasize different properties of the image like pixel intensities, texture, color, *etc*. Some of the local image description techniques are *distribution based*, in which histograms are used to represent different characteristics of appearance or shape. Example of these techniques are spin images [71], the popular Scale Invariant Feature Transformation (SIFT) [92] and its extension GLOH (Gradient Location Orientation Histogram) [101]. Also, geometric histograms [17] and shape context descriptors [23] belong to this type of descriptors.

Some techniques are *spatial-frequency based*, in which the frequency content of the images are described. Example of these descriptors are Fourier transform and Gabor filters [150].

Some descriptors depend on *image derivatives* computed up to a given order to approximate a point neighborhood (differential descriptors). Example of these differential descriptors are the local jets [75] and steerable filters [49]. Other techniques are developed such as moment invariants [149]

All previously mentioned descriptors are computed form intensity values of the images, while the use of local color-based descriptors has received little attention by most of the approaches. However, some local color descriptors are developed such as in [103] where local descriptors based on color moments are introduced. Furthermore, in [95] invariant signature based on the modes in the local color histogram is proposed.

Authors in [148] presented a set of local color descriptors with different criteria such as photometric robustness, geometric robustness, photometric stability and generality. Moreover, new different color descriptors such as rgb-SIFT and RGB-SIFT are recently proposed and evaluated in [147].

Representation of Range Images

Local Keypoints Local representation using local keypoints is an emerging technique for recognition from range images. How the local keypoints are selected is different from approach to another. Some approaches such as [53, 70] used random selection of points to compute the surface descriptors. Chen and Bhanu [30] selected keypoints by considering points with high curvature. Another technique is developed by Li and Guskov [87, 88], where they detected salient points by building a scale space representation similar to [91] of the 3D surface.

Another different technique for detecting keypoints in range images is presented by Roth [127]. He used feature points extracted from intensity images corresponding to the range data. The 3D points associated with these feature points extracted from the

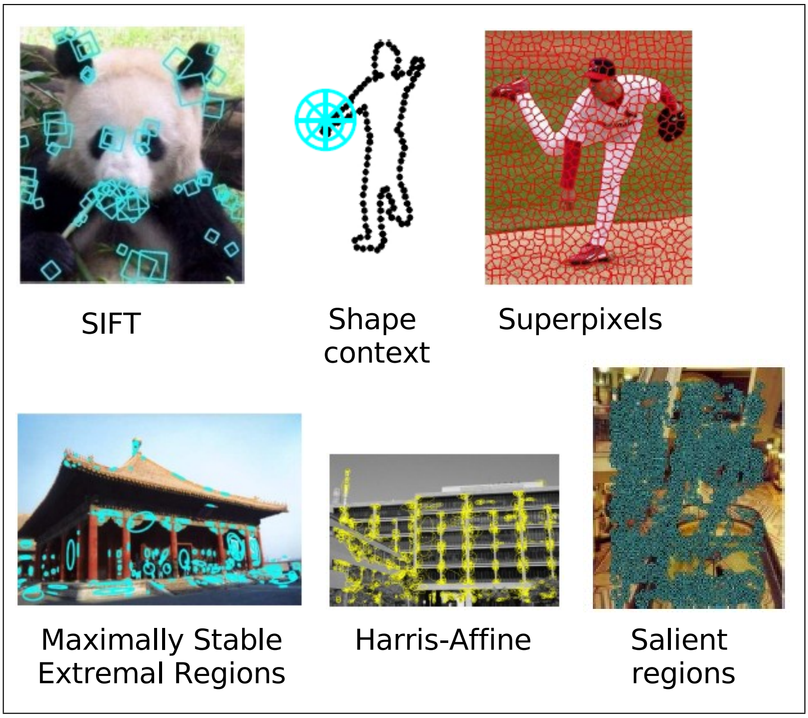


Figure 2.1: *Example of local representations of 2D images*: SIFT descriptor [91] Shape context descriptor [23], Superpixels [125], Maximally Stable Extremal Regions (MSER) detector [94], Harris-Affine detector [99] and Salient regions [73]

intensity images are then used to find correspondence between two range images.

Local Descriptors There are different local descriptors for range images in the literature. The most famous one is the Spin Images presented in [70]. Spin Images are extracted at each oriented point (a point along with its normal) of the object's surface and are a 2D histogram of the cylindrical coordinates of its surrounding points.
Another descriptor is the Point Signatures [32], which is a one dimensional signature that describes the surface surrounding a point. Point Signatures is invariant to rotations and translations and can, therefore, be used to establish correspondence between two different views of an object [97].
Another descriptor is Surface Signatures presented in [159], which are images of the

surface curvature information seen from certain point. Other local shape descriptors include PCA-based descriptors [141], [96], Regional Point Descriptors [53], Points Fingerprints [139] and local feature histograms [63].

2.2 Learning Objects Representations

Learning is the next important step, after representation, in any GOR system. Visual object categories are learned by the model from the representations of a number of examples (training images or training sequences). Learning can be performed in a supervised or an unsupervised manner. In the *supervised* learning, the labels of the training images as well as bounding boxes denoting the object locations in the images are given to the learning model. In the *unsupervised* learning, the learning model does not have any information about the different object classes but the given training images. No image labels or any other information concerning the object locations within the images is given. For the supervised learning, a further division into different levels is given by Opelt [111] as follows:

- **Weakly supervised**: is a level between supervised and unsupervised learning. Only labels of the training images are given to the learning model. No further information about objects, such as their locations within the images, is given.

- **Highly supervised**: is one level after the supervised learning. Beside the labels of training images and the bounding boxes of object in them, more information is given to the learning model by segmenting the objects from the training images.

- **Completely supervised**: is one level after the highly supervised learning and include the information given to the model in the highly supervised learning, in addition to another information by the user such as additional effort during learning.

Many different learning techniques are used in the literature for GOR. For example, Lowe used a nearest-neighbor algorithm [91] while Agarwal and Roth [13] used Winnow. The Expectation-Maximization (EM) algorithm is been used successfully by Dorko and Schmid [37] for learning a generative probabilistic objects model.
Support Vector Machines (SVM) has received the attention of many researchers and is been used successfully in many models such as [52, 29, 105, 108]. Boosting is used successfully in many recognition models such as [151, 113, 111, 162, 64, 15].

Following, we give a brief explanation to Boosting and how it works [2]. Afterwards, we give an overview of the boosting algorithms used throughout this thesis.

2.2.1 Learning with Boosting

Boosting is the underlaying learning technique in the recognition approaches mentioned in this thesis. Boosting algorithms are currently among the most popular and most successful algorithms for pattern recognition applications such as feature selection [140], face detection [151] and generic object recognition [114, 58]. The underlaying idea of boosting algorithms is to construct a ”strong” classifier using only a training set and a ”weak learning” algorithm. A ”weak hypothesis” produced by the weak learning algorithm has a probability of misclassification that is slightly below 50% (performs better than random guess). A ”strong hypothesis” has a much smaller probability or error on test data. Hence, boosting algorithms boost the weak learning algorithm to achieve a strong hypothesis.

In order to exploit the advantage of the weak learning algorithm over random guessing, the data is re-weighted before training the weak learning algorithm in each iteration. The algorithm increases the weights of the examples that are wrongly classified by the weak hypothesis. The end result is a final strong hypothesis given by a thresholded linear combination of the weak hypotheses [128].

There are many boosting algorithm where the main variation among many of them is the method of weighting training data points and hypotheses. AdaBoost is very popular and perhaps the most historically significant, as it was the first algorithm that could adapt to the weak learners. However, there are many more recent algorithms such as LPBoost [35], TotalBoost [155], SoftBoost [154], GentleBoost [51], joint Boosting [144] and many others. Following, we will provide brief discussions to the boosting algorithms used by our approaches. Discussions to different boosting algorithms could be found in the given references. However, before presenting the boosting algorithms, preliminaries of boosting are given.

Preliminaries of Boosting

In the boosting settings, a set of N labeled training examples (I_i, l_i) for $i = 1 \ldots N$ are given, where the instances I_i are in some domain χ and the labels $l_i \in \pm 1$. Boosting

---

[2] Boosting the underlaying learning algorithm in all the recognition models presented in the thesis.

algorithms maintain a distribution **w** on the examples N such that the hard to classify examples receive more weight. The boosting algorithm is run for a certain number of iterations T. In each iteration, the algorithm gives the current distribution to a weak learning algorithm (*weak learner*), which returns a new weak hypothesis $h : \chi \rightarrow [-1, 1]^N$ with a certain guarantee of performance.

One measure of the performance of a weak hypothesis h with respect to distribution **w** is its *edge*, $\gamma_h = \sum_{i=1}^{N} w_i l_i h\left(I_i\right)$. When the range of h is ± 1 instead of the range of $[-1, 1]$, the edge is just an affine transformation of the weighted error ϵ_h of hypothesis h: i.e. $\epsilon_h\left(\mathbf{w}\right) = \frac{1}{2} - \frac{1}{2}\gamma_h$. A hypothesis that predicts perfectly has an edge $\gamma = 1$ while a hypothesis that always predicts incorrectly has an edge $\gamma = -1$. A random hypothesis has an edge $\gamma = 0$. The higher the edge, the more useful is the hypothesis for classifying the training examples. The edge of a set of hypotheses is defined as the maximum edge of the set.

After a hypothesis is received, the algorithm must update its distribution **w** on the examples. Boosting algorithms (for separable case) commonly update its distribution **w** by placing an edge constraint on the most recent hypothesis. Such algorithms are called *corrective* [124, 154]. In *totally corrective* algorithms, the distribution is updated to have a small edge with respect to all of the previous hypotheses [155, 154]. The final output of the boosting algorithm is always a convex combination of weak hypotheses $f\left(I_i\right) = \sum_{k=1}^{T} \alpha_k h_k\left(I_i\right)$, where h_k is the hypothesis added at iteration k and α_k is its coefficient [154]. The *hard margin* of a labeled examples (I_i, l_i) is defined as $\rho_i = l_i f\left(I_i\right)$. The margin of a set of examples is taken to be the minimum margin of the set.

It is convenient to define a N-dimensional vector $\mathbf{u}^m$ that combines the weak hypothesis h_m with the label l_i of the N examples: $u_i^m = l_i h_m(I_i)$. With this notation, the edge of the k-th weak hypothesis becomes $\mathbf{w}.\mathbf{u}_m$ and the margin of the n-th example with respect to a convex combination of the first $k-1$ hypothesis is $\sum_{m=1}^{k-1} u_n^m \alpha_k$ [154].

Now, the Boosting algorithms used throughout this thesis are briefly presented with using the same notations used for describing the boosting preliminaries.

AdaBoost Algorithm

AdaBoost (Adaptive Boosting) algorithm introduced by Freund and Schapire [50] is the most well known boosting algorithm. It is considered as the first step toward more practical boosting algorithms . AdaBoost is adaptive, in that the linear coefficient of the weak hypothesis depends on the weighted error of the weak hypothesis at the time when the weak hypothesis is added to the linear combination. In iteration k, the weight

w_i is decreased if the prediction for I_i was correct $(h_k(I_i) = l_i)$, and increased if the prediction was incorrect. Algorithm 1 gives an overview of the AdaBoost algorithm.
The AdaBoost algorithm presented by Freund and Schapire [50] produces a weak hypothesis h in the form: $h : \chi \rightarrow \{-1, 1\}$ *e.g.* a strict classifier that maps the input to $\{-1, 1\}$ without giving any prediction confidence. Therefore, the AdaBoost algorithm of Freund and Schapire [50] is also called *Discrete AdaBoost*.
Schapire and Singer [133] extended the framework of the AdaBoost algorithm presented in [50] in which each weak hypothesis generates not only predicted classifications, but also self-rated confidence scores, which estimate the reliability of each of its predictions. Schapire and Singer introduced in [133] the AdaBoost with confidence-rated prediction algorithm (*i.e.* Real Adaboost). It differs from the AdaBoost [50], as mentioned, in that the weak learner of the first computes a weak hypothesis $h : \chi \rightarrow \mathbb{R}$. The sign of h is interpreted as the predicted label (-1 or +1) to be assigned to the instance I_i and the magnitude $\mid h(I_i) \mid$ as the confidence of this prediction. Moreover, the method of computing the coefficient of the weak hypothesis α is different [133]. Algorithm 2 displays the AdaBoost with confidence-rated prediction algorithm. Further details of the algorithm could be found in [133].
The AdaBoost algorithm has two interesting properties [124]. First, in its (discrete) version [50], the training error can be reduced exponentially as the number of weak hypotheses increases. If the weighted training error of the k-th weak hypothesis is $\epsilon_k = \frac{1}{2} - \frac{1}{2}\gamma_k$, then the upper bound on the training error of the signed linear combination is reduced by factor $1 - \frac{1}{2}\gamma_k^2$. Second, it has been experimentally observed that AdaBoost continues to learn even after the training error of the signed linear combination is zero [132]. This is because the margins of the training examples continue to increase even after the training error is zero. However, the algorithm suffers from some limitations. Following, a discussion of these limitations as well as a presentation of boosting algorithms which try to overcome these limitations are given.

SoftBoost Algorithm

AdaBoost algorithm has the advantage of generating combined hypotheses with large margins and works well on data with low noise [132]. However, studies showed that the performance of AdaBoost is affected with the the presence of high noisy data [154]. Some studies showed that, in such case, a large margin on all training data can not be achieved without affecting the generalization performance, as AdaBoost concentrates too much on outliers and hard to classify examples [132, 154].
Due to this reason, many variants of AdaBoost appeared to cope with this problem and

```
Input: $S = \langle (I_1, l_1), ..., (I_N, l_N) \rangle; I_i \in \chi, l_i \in \{-1, +1\}$.
Initialize: $w_1(i)$ to the uniform distribution.
for $k = 1, ...T$: do
    (a) Train weak learner using distribution $\mathbf{w}_k$.
    (b) Get weak hypothesis $h_k : \chi \rightarrow \{-1, +1\}$.
    (c) Calculate the classification error as:
        $\varepsilon_k = \frac{\sum_{i=1}^{N} (h_k(I_i) \neq l_i) w_i}{\sum_{i=1}^{N} w_i}$
    (d) Choose $\alpha = \varepsilon_k (1 - \varepsilon_k)$.
    (e) Update:
        $w_{k+1}(i) = \frac{w_t(i) exp(-\alpha_k l_i h_k(I_i))}{Z_k}$
    where $Z_k$ is a normalization factor (chosen so that $w_{k+1}$ will be a
    distribution).
end
Output: Final hypothesis: $H(I) = sign(\sum_{k=1}^{T} \alpha_k h_k(I))$.
```

Algorithm 1: AdaBoost algorithm [50].

to trade off the number of margin errors and the size of the margin. This is achieved by restricting the weighting maintained by the algorithm to not concentrate too much on the most difficult (hard to classify) examples [154]. Examples of these algorithms are AdaBoost with soft margin [123] and LPBoost [35].

On the other hand, and as previously mentioned, the hypotheses combination produced by AdaBoost has a large margin on the data. This margin is not necessarily the maximum hard margin. Therefore, many new versions of AdaBoost, which try to provide a maximum hard margin, have been developed such as AdaBoost* [124], TotalBoost [155], and many other algorithms [154]. However, such algorithms are not suitable for real-world applications with noisy data as over-fitting is more problematic for them than the original AdaBoost algorithm [154].

SoftBoost is a newly presented boosting algorithm [154], which combines the previously mentioned two lines of research in a single algorithm, that it implements the *soft margin* idea in a practical boosting algorithm.

SoftBoost is a *totally corrective* algorithm which optimizes the soft margin and tries to

```
Input: $S = \langle (I_1, l_1), ..., (I_N, l_N) \rangle$; $I_i \in \chi$, $l_i \in \{-1, +1\}$.
Initialize: $w_1(i)$ to the uniform distribution.
for $k = 1, ...T$: do
    (a) Train weak learner using distribution $\mathbf{w}_k$.
    (b) Get weak hypothesis $h_k : \chi \rightarrow \mathbb{R}$.
    (c) Choose $\alpha \in \mathbb{R}$.
    (d) Update:
    $w_{k+1}(i) = \frac{w_t(i) exp(-\alpha_k l_i h_k(I_i))}{Z_k}$
    where $Z_k$ is a normalization factor (chosen so that $w_{k+1}$ will be a distribution).
end
Output: Final hypothesis: $H(I) = sign(\sum_{k=1}^{T} \alpha_k h_k(I))$.
```

Algorithm 2: AdaBoost with confidence-rated predictions algorithm [133].

produce a linear combination of hypotheses with the maximum one [154]. The term ”soft” here means that the algorithm does not concentrate too much on outliers and hard to classify examples (*e.g.* as in AdaBoost). It allows them to lie below the margin (i.e. to have wrong predictions) but penalizes them linearly via slack variables. Figure 2.2 gives an explanation to the difference between hard and soft margins.

Therefore, it seems that SoftBoost avoids the problem of over-fitting exist in AdaBoost when using training data with high degree of noise. A brief description of the SoftBoost algorithm is given below (see also Algorithm 3). Further details about the algorithm could be found in [154].

SoftBoost takes as input, a sequence of examples $S = \langle (I_1, l_1), ..., (I_N, l_N) \rangle$ in addition to an accuracy parameter δ and a capping parameter ν (see algorithm 3). This capping parameter specifies how many examples could be mistrusted or, in other words, how many examples are allowed to lie below the margin. The algorithm has a weak learner which provides a hypothesis with an edge with a known guarantee g. The initial distribution $\mathbf{w}^0$ of the algorithm is uniform. In each iteration k, the algorithm prompts the weak learner for a new weak hypothesis, adds it into the constraints set, and updates its distribution $\mathbf{w}^{k-1}$ to $\mathbf{w}^k$ by minimizing the relative entropy $\Delta\left(\mathbf{w}, \mathbf{w}^0\right) := \sum_n \left(w_n \ln \frac{w_n}{w_n^0} \right)$ subject to the constraints:

$$\mathbf{w}^{k+1} = \operatorname*{argmin}_{\mathbf{w}} \Delta\left(\mathbf{w}, \mathbf{w}^{0}\right) \tag{2.1}$$

$$\text{s.t. } \mathbf{w} \cdot \mathbf{u}^{m} \leq g - \delta, \quad \text{for } 1 \leq m \leq k,$$

$$\sum_{n} w_n = 1, \ \mathbf{w} \leq \frac{1}{\boldsymbol{\nu}} \mathbf{1}$$

This optimization function could be easily solved with vanilla sequential quadratic programming methods (see [155] for details). The relative entropy in the objective assures that the probabilities of the examples are always proportional to their exponentiated negative soft margins (not shown). In other words, more weight is put on the examples with low soft margin, which are the examples that are hard to classify [154].

AdaBoost and SoftBoost algorithms are used for binary classification of object classes. For the multi-class learning and classification task, several boosting algorithms exist such as AdaBoost.M1, AdaBoost.M2 [50] and JointBoosting [144] algorithms. Following, we dicuss briefly the Joint Boosting algorithm, as it is used in our multi-class recognition model presented in chapter 7.

Joint Boosting Algorithm

Joint Boosting algorithm is a multi-class boosting algorithm with feature sharing developed by Torralba *et al.* [144]. The idea of the Joint boosting algorithm is that at each boosting iteration, the algorithm examines various subsets of classes $S_n \subseteq C$ (a total of $2^C - 1$ possible subsets) that will share features, where C is the total number of classes to be learned. The algorithm considers fitting a weak classifier (hypothesis) to distinguish the subset from the background (set of other classes). The subset that maximally reduces the error on the weighted training set for all classes is chosen by the algorithm. The best weak learner $h(I, c)$ is then added to the strong learners $H(I, c)$ for all the classes $c \in S_n$ and their weight distributions are updated. Joint Boosting algorithm is based on the GentleBoost algorithm [51] which is modified to suit the multi-classification case as well as the feature sharing property. Algorithm 4 gives a summarization of the Joint Boosting algorithm. Further details of the algorithm could be found in [144].

```
Input: $S = \langle (I_1, l_1), ..., (I_N, l_N) \rangle$, desired accuracy $\delta$, and capping
        parameter $\nu \in [1, N]$.
Initialize: $w_n^0$ to the uniform distribution.
for $k = 1, ...$ do
    (a) Send $\mathbf{w}^{k-1}$ and $\{\mathbf{u}_1, ..., \mathbf{u}_{k-1}\}$ to the weak learner and obtain
    hypothesis $h^k$ which has edge at least $g$ w.r.t. $\mathbf{w}^{k-1}$.
    Set $u_n^k = h^k(I_n) l_n$.
    (b) Update
        $\mathbf{w}^k = \underset{\mathbf{w}}{\operatorname{argmin}} \Delta(\mathbf{w}, \mathbf{w}^0)$
        s.t. $\mathbf{w} \cdot \mathbf{u}^m \leq g - \delta, \quad \text{for } 1 \leq m \leq k,$
        $\sum_n w_n = 1, \ \mathbf{w} \leq \frac{1}{\nu} \mathbf{1}$
    (c) If above infeasible or $\mathbf{w}^k$ contains a zero then $T = k - 1$ and
    break.
end
Output: $f_{\mathbf{W}}(x) = \sum_{m=1}^{T} \mathbf{W}_m h^m(I)$,where the coefficients $\mathbf{W}_m$
        maximize the soft margin over the hypotheses set $\{h^1, ..., h^k\}$
        using the LP problem (1) in [154].
```

Algorithm 3: SoftBoost with accuracy parameter δ and capping parameter ν [154].

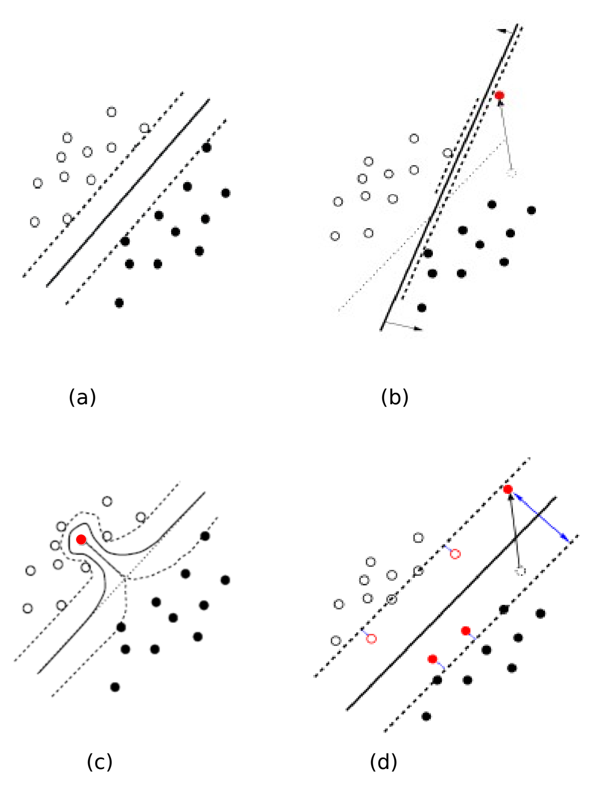


Figure 2.2: *Hard vs. Soft margins*. (a) Finding a maximum hard margin ”hyperplane” on reliable data. (b) On data with outliers. (c) On data with mislabeled example. The solid line shows the resulting decision boundary, whereas the dashed line marks the margin area. In (b) and (c), the original decision boundary is plotted with dots. The hard margin implies noise sensitivity, as only one example can spoil the whole estimation of the decision boundary [123]. (d) Soft margin on data with outliers. Hard to classify examples, e.g. outliers and mislabeled examples, are allowed to lie below the margin (have wrong prediction) but are penalized linearly via slack variables.

```
1) Initialize the weights $w_i^c = 1$ and set $H(v_i, c) = 0$, $i = 1 \ldots N$,
   $c = 1 \ldots C$.
2) Repeat for $k = 1, 2, \ldots T$
    a) Repeat for $n = 1, 2, \ldots 2^C - 1$
        i) Fit shared stump:
            $$h_k^n(v_i, c) = \begin{cases} \text{as} & \text{if } v_i^f > \theta \text{ and } c \in S(n) \\ \text{bs} & \text{if } v_i^f < \theta \text{ and } c \in S(n) \\ \text{ks} & \text{if } c \notin S(n) \end{cases}$$
        ii) Evaluate the error
            $$J_{wse}(n) = \sum_{c=1}^{C} \sum_{i=1}^{N} w_i^c (z_i^c - h_k^n(v_i, c))^2$$
   .
    b) Find best subset: $n^* = argmin_n J_{wse}(n)$.
    c) Update the class estimates: $H(v_i, c) := H(v_i, c) + h_k^{n^*}(v_i, c)$.
    d) Update the weights: $w_i^c := w_i^c e^{-z_i^c h_k^{n^*}(v_i, c)}$.
```

Algorithm 4: Joint boosting algorithm [144]. v_i^f is f'th feature of the i'th training example, $z_i^c \in \{-1, +1\}$ are labels of class c and w_i^c are the unnormalized example weights. N is the number of examples and T is the number of boosting iterations.

2.3 Classification

After the object representations are learned by the learning algorithm, classification or recognition of new instances of the objects or object classes is the next step to be accomplished. The classification (recognition) can be formulated as follows [118]: "Given a number of learned categories, a new images should be processed and presented and a decision should be derived, whether a known category appears in the data or not."

Parametric techniques try to find a model (e.g. probabilistic model) which is estimated from the training images. Some nonparametric techniques work directly on the feature space. These methods are required when it is needed to deal with distributions in features space that are hard to be modeled explicitly. They are also needed when dealing with highly overlapping parametric models is required [118]. For more readings, the following reference could be helpful [38].

2.4 GOR Datasets

For a recognition model to be tested, a suitable dataset is needed. In the last years, many different datasets have appeared for evaluating and comparing different GOR approaches and models. For a GOR dataset, several aspects should be considered. These aspects can be summarized as follows:

- Number of images per category: they should provide many images per each category.
- Intra-class variability: they should cover the high intra-class variabilities exist among objects within the same visual class.
- Inter-class variabilities: also, they should have examples of low inter-class variabilities among different classes (examples of different object classes which are visually similar).
- Ground truth data: they should provide the images labels. Moreover, a contour or a bounding box can also be provided for objects localization.
- Number of categories: they should provide images of many different object categories.

However, available datasets vary with respect to the previously mentioned different aspects, in addition to the following:

- Viewpoint change: some datasets provide different arbitrary viewpoints of the objects of the different classes while other datasets provide viewpoints at very controlled angels. Some datasets provide only one certain aspect of an object class.

- Background clutter: the datasets vary from providing no background clutter at all and notably seen objects to real world scenes where the object covers only a small region in the image.
- Scale variation: the scale of objects are varied in the available datasets from none, small or high change in scale.

The choice of a suitable dataset depends on the task. However, a good GOR model should deliver good performance on all datasets. Actually, this is not achieved yet and researchers are trying to make their approaches deal with complex cases in a progressive way.

Now, we will mention some of the famous datasets used for object recognition either using 2D (2D image dataset) or range images (range image datasets). For the 2D datasets, we mention those which are used as a benchmark by almost all state-of-the-art approaches for building and evaluating their GOR models. For the 3D datasets (range datasets), we mention those used by the state-of-the-art approaches for specific object recognition [3].

2.4.1 2D Images Datasets

ETH-80

A dataset of 8 different categories, each category contains images of 10 different object instances [1] . The images of the dataset are acquired in controlled settings where the objects are placed on a turntable in front of an uncluttered blue background and images are taken from 42 equally spaced viewpoints around a a half view-sphere.

Figure 2.3 shows example images of the dataset. We do not report any results using this dataset as it is too simple for our purpose. However, evaluation results using the dataset are given in [82, 84].

Caltech

A popular and widely used dataset acquired and published by Caltech [2] and first used by Fergus *et al.* [43]. The dataset contains 5 different object classes: cars-rear, motorbikes, airplanes, faces and leaves. Additionally, a background class is also presented. The different objects are shown in almost the same position in the images, with little

---

[3] All available range image datasets are useful for the specific 3D object recognition task only and there is no available range dataset suitable for GOR.

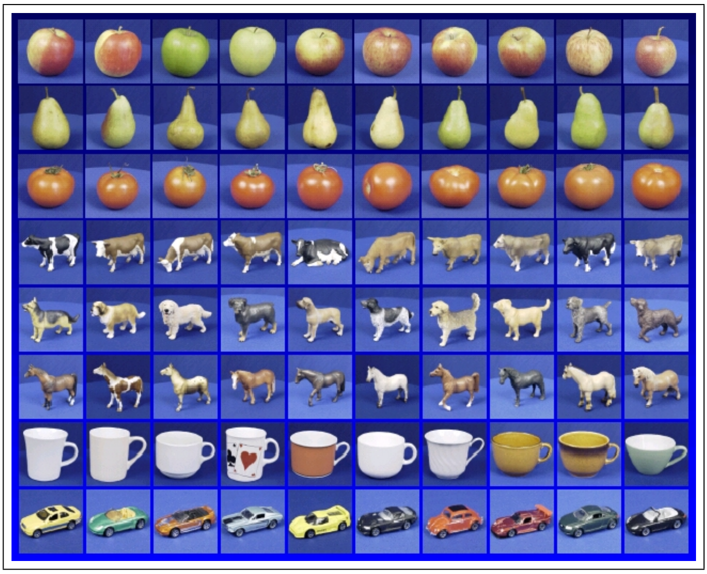


Figure 2.3: Example images of the different object classes of ETH-80 dataset.

or no background and from very similar viewpoints. Figure 2.4 displays example images of the mostly used object classes of the dataset (Caltech 4). Because the dataset is been used by many different approaches, many results are exist in the literature for comparison. Therefore, we evaluate our model for 2D GOR using this dataset (Caltech 4).

UIUC

The (University of Illinois Urbana Campaign) UIUC dataset [3] has only one object class, side views of cars (car-side) in addition to background images. The dataset is acquired by Agarwal *et al.* [13] and example images are shown in figure 2.5. The complexity of the dataset is average and suffers from the problems of the Caltech dataset (little background clutter and pose variations) [119]. Therefore, we do not report any recognition results using this dataset.

Graz

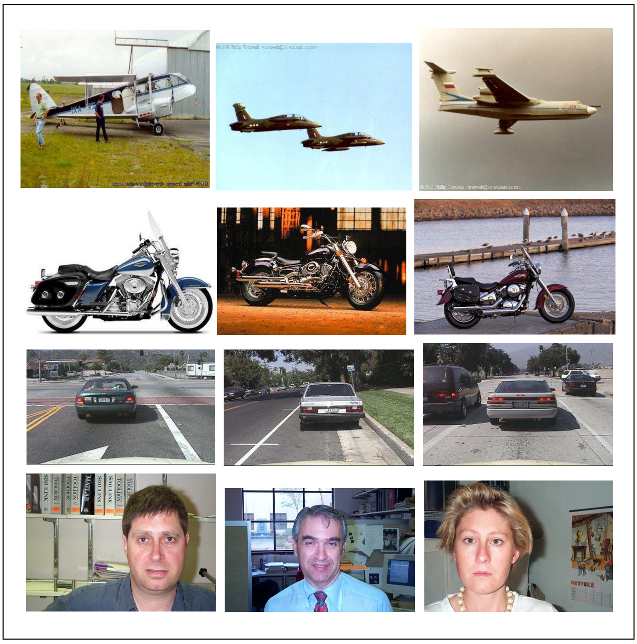


Figure 2.4: Example images of the different object classes of Caltech 4 dataset.

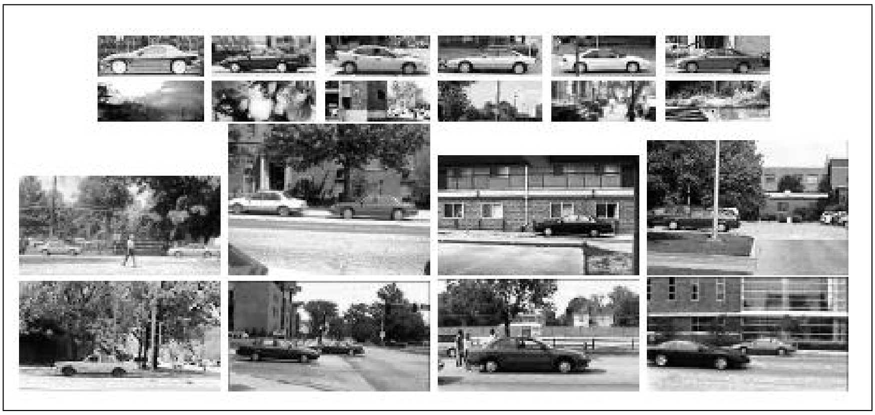


Figure 2.5: Example images of the different object classes of UIUC dataset.

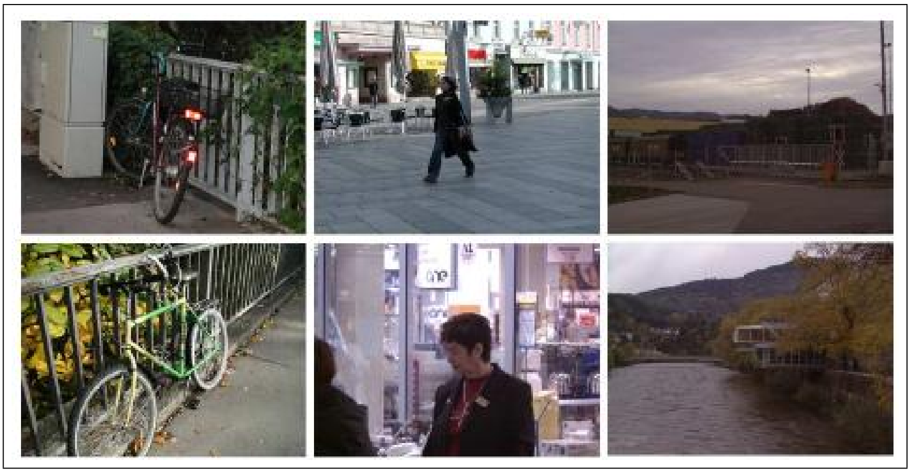


Figure 2.6: Example images of the different object classes of Graz01 dataset.

Graz01 Contains 2 different object classes, namely bikes and persons [4], in addition to a background class (no bikes-no persons). The images are highly complex with high variability in scale, viewpoint and illumination (see Figure 2.6). Evaluations using the dataset are given in [112, 114]

Graz02 Contains 3 different object classes: cars, bikes and persons [5]. Additionally, a background class is constructed. Example images are shown in figure 2.7. The images of the dataset is more complex than those of Graz01 dataset. Moreover, the appearance of the background of the images are balanced, so that similar context is shared by the different object classes, including the background class. Evaluations of different recognition model is done using this dataset such as in [114, 104]. Also, we evaluate our model for 2D GOR using this dataset.

2.4.2 A Dataset for 3D Object Categories

The dataset was presented by Savarese and Fei-Fei in [130]. It consists of 8 object categories (bike, shoe, car, iron, mouse, cell phone, stapler, toaster) [6]. For each object category, the dataset contains images of 10 individual object instances under 8 viewing angles, 3 heights and 3 scales for a total number of 7000 images. Images are

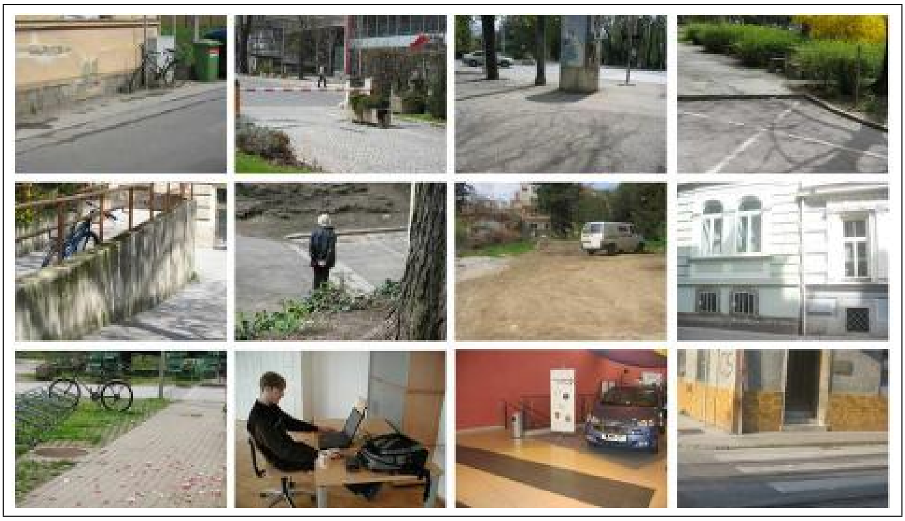


Figure 2.7: Example images of the different object classes of Graz02 dataset.

roughly 400 × 300 pixels in bmp format. Example images are displayed in Figure 2.8. The dataset is been constructed to be used for generic 3D object recognition.

2.4.3 Range Images Datasets

OSU/SAMPL Range Images Dataset

The dataset [7] contains images which are acquired by several real range sensors from different sources, in addition to synthetic data (see figure 2.9). The images of the dataset are available in one or both of the formates: grayscale GIF and a compressed neutral format ('txt' format) with fixed point measurements in X, Y and Z. The dataset is useful for building and evaluating specific 3D object recognition. therefore, do not report any evaluation results using the dataset.

Stuttgart Range Images Dataset

A popular range image dataset [8]. It is a collection of 9720 synthetic range images of 30 free form objects as shown in figure 2.10. The range images are taken from high resolution polygonal models. The dataset is suitable for specific 3D object recognition

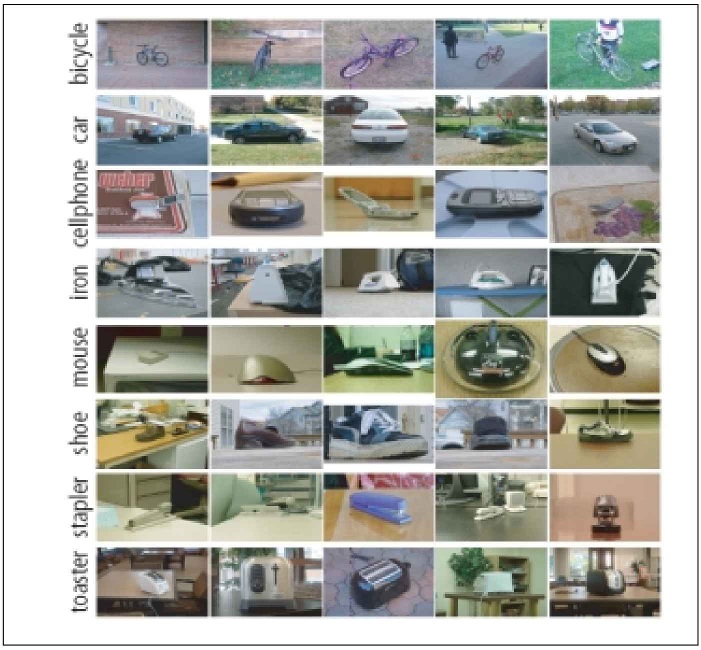


Figure 2.8: Example images of the different object classes of the 3D Object Categories dataset of Savarese and Fei-Fei in [130].

from range images and has been used for evaluating different recognition models such as [63, 88].

2.5 Evaluation of Recognition

Evaluating the performance of the recognition model is an important issue in recognition. Given a classifier and a set if images, there are four possible classification outcomes, which are mentioned and defined as follows:

- True Positive (TP): an image with positive label is been classified as positive.
- True Negative (TN): an image with negative label is been classified as negative.
- False Positive (FB): an image with negative label is been classified as positive.

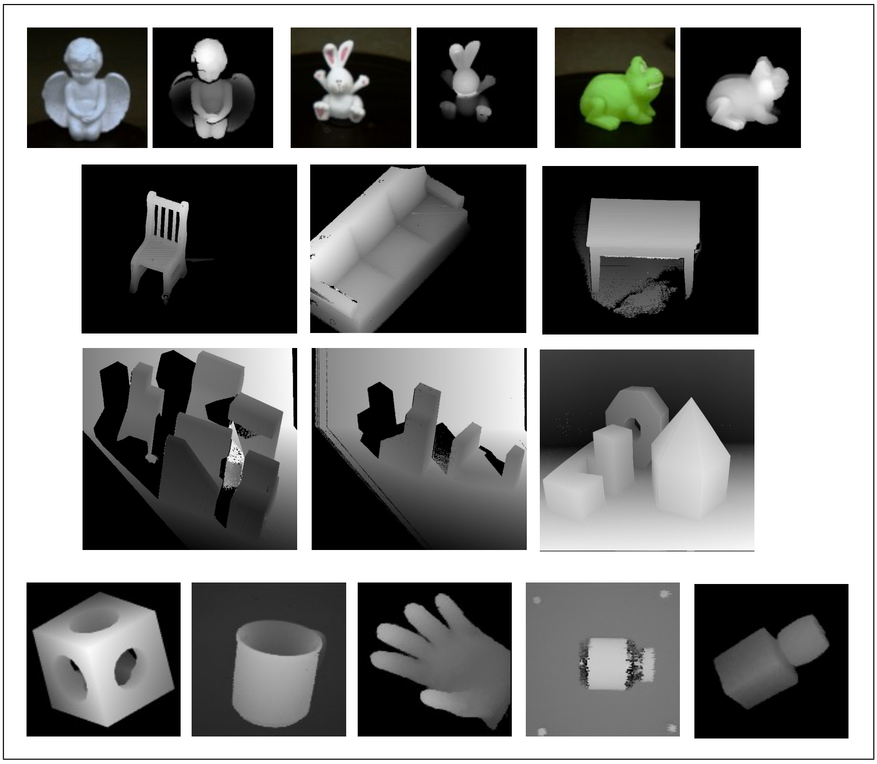


Figure 2.9: Example images of OSU/SAMPL range dataset.

- False Negative (FB): an image with positive label is been classified as negative.

The performance of a classifier can then be measured with different methods. The common method is the *recognition rate* (i.e. accuracy), which is simply the percentage of the correctly classified images. With the help of the previously mentioned outcomes, it can also be defined as follows:

$$\text{Recognition rate} = \frac{\#\text{TP} + \#\text{TN}}{\#\text{P} + \#\text{N}} \tag{2.2}$$

where $\#$ denotes number of images. Moreover, the classification performance can be also measured using the *Receiver-Operating-Characteristics curve (ROC)* [40] with:

$$\text{True Positive Rate (TPR)} = \frac{\#\text{TP}}{\#\text{P}} \tag{2.3}$$

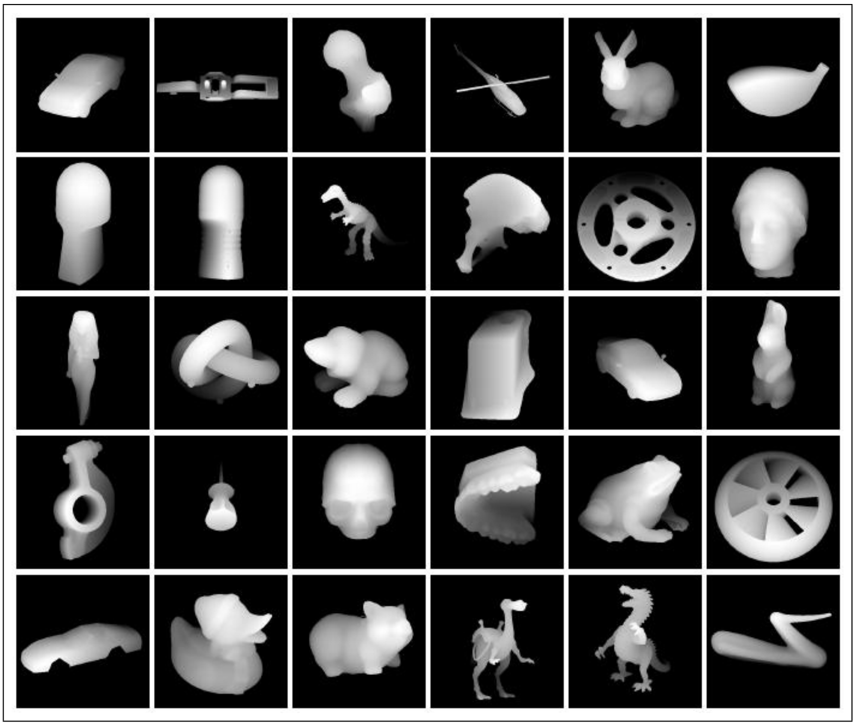


Figure 2.10: Example images of Stuttgart range dataset.

and

$$\text{False Positive Rate (TPR)} = \frac{\#\text{FP}}{\#\text{N}} \tag{2.4}$$

Figure 2.11(a) gives an example of ROC curves. Two values are extracted from the curve and are important for the performance measure. The first value is the *ROC-equal-error rate* and is defined as the point on the ROC curve where the true positive rate = 1-false positive rate [118]. It gives a nice trade-off between the true positive and false positive. The second value is the area under the ROC curve (ROC-AuC) which is useful when comparing the performance of two classifiers is required. More detailed information concerning the ROC curves could be found in [40].

While the ROC curves is a good measure for the discrimination ability in recognition tasks, the *Recall-Precision Curve (RPC)* is more suitable for measuring the localization, especially for object detection models that use local patches [118] (see figure 2.11(b)). The task is then not only classifying the images but to decide, whether a certain local patch belongs to the object or not. Recall and precision can be defined as follows:

$$\text{recall} = \frac{\#\text{TP}}{\#\text{P}} = \text{TPR} \tag{2.5}$$

and

$$\text{precision} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}} \tag{2.6}$$

The RPC curves plots recall against (1-precision), which is defined as follows:

$$\text{1-precision} = \frac{\#\text{FP}}{\#\text{TP} + \#\text{FP}} \tag{2.7}$$

Another important measure for the classification performance in the multi-class classification case is the *confusion matrix*. The confusion matrix represents in each of its row how many examples of one class were classified to belong to other classes. Recognition is perfect when there are only entries in the main diagonal of the matrix. Numbers in entries other than the diagonal indicate that a certain category tends to be confused with another.

2.6 Conclusions

This chapter has presented the main issues that should be put into consideration when researchers build a model or system for GOR. These issues are the representation of object categories, learning these representation and classification of new instances. However, these issues are considered the building blocks for any recognition model in general. The choice of each issue is crucial for the finale performance of the recognition model. Moreover, datasets are important for providing data for building and evaluating any recognition model. This chapter has presented an overview of the current available datasets used for GOR. The choice of a suitable dataset is very significant for building the GOR model. The use of simple datasets in evaluating the recognition model could

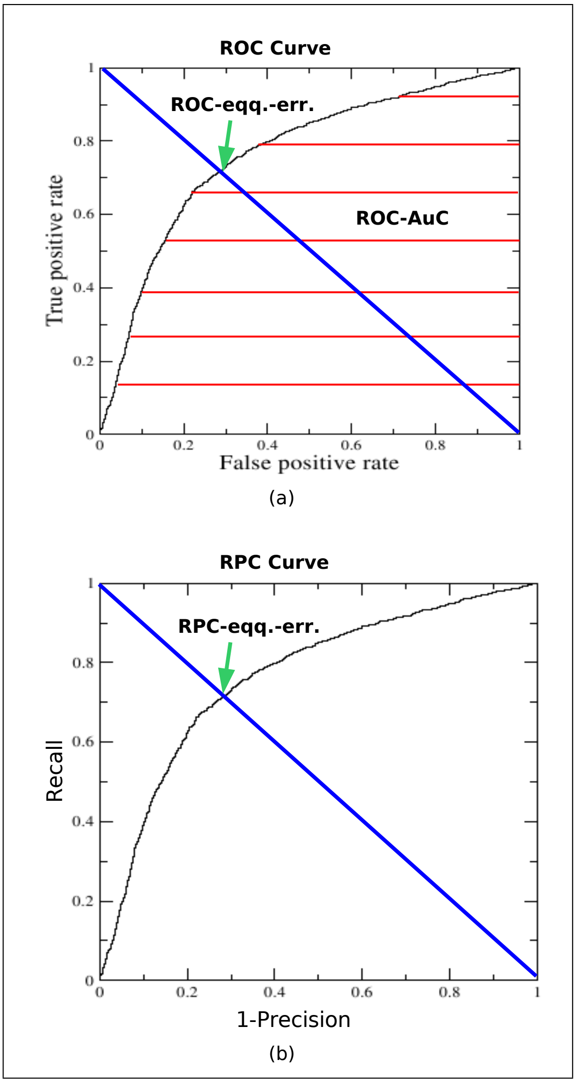


Figure 2.11: *ROC and RPC curves.* (a) Receiver-Operating-Characteristics curve (ROC) with the performance measures, the ROC-equal-error rate and the ROC-AuC. (b) Recall-Precision Curve (RPC) and the corresponding RPC-equal-error rate.

yield good performance, but will not reflect the actual performance of the GOR model or its suitability for solving the problem. The choice of the dataset and its effect on the recognition will be further discussed in chapters 4 and 5. Finally, evaluating the recognition model using adequate measurements is an essential step. Different important performance measurements for GOR have been presented in this chapter.

Chapter 3

Related Work

There has been fairly a large amount of work in the domain of object recognition. In the 1960's, began the first trails of object recognition. Consequently, a large and extensive body of literature on the problem has appeared . Over the years since the first trails, progress in handling the problem has been achieved, which can be judged by the amount of realistic data and conditions used in recognition experiments.

Range data was first used in recognition in 1970's since it presents direct information about the 3D environment. Afterwards, in 1980's, intensity images were used directly. However, the used images presented the objects with uniform background and without occlusion in order to simplify the segmentation and recognition processes. The use of natural images was then addressed by different methods which recognize only a single object instance from different viewpoints.

In 1980's, the first work on generic object recognition (GOR) appeared, dealing with a limited set of classes such as faces and digits, usually in constrained environment . Afterwards, from the late 1990's, the work in GOR started to undertake a wider variety of classes in more natural image environment.

This chapter presents an overview of the different approaches and models that are developed to tackle the object recognition problem in general and specifically the problem GOR. An overview of the approaches developed to deal with the specific object recognition is presented in section 3.1. Then, section 3.2 sheds light on the models and state-of-the-art approaches developed to tackle the difficult problem of GOR [1].

---

[1] It should be noted that some of the GOR approaches were not yet published when we started the work in our approaches. Some are published at the same time or afterwards.

3.1 Specific Object Recognition

Large amount of literature can be found on specific object recognition. Specific object recognition sets up the general basis for generic object recognition. For this reason, we start our literature review by providing an overview of the main models and approaches developed for the task of specific object recognition. In our review, we divide the approaches into approaches which are based on shape and geometric information of objects and those which make use of objects appearance information.

3.1.1 Geometrical and Model-based Approaches

The early object recognition systems were model-based. One of the first general purpose vision systems that performed object recognition was SRI vision module presented by Agin [14]. SRI vision system used binary images and it was based on connectivity analysis, which is a procedure that breaks a binary image into its connected components. The connectivity program extracts information about the blob that will be used later on, such as the maximum of its extent, area, perimeter and coordinates of the points on the perimeter, while extracting connected components. The SRI vision system recognized objects with two ways: the nearest neighbor technique and the binary decision tree procedure.

Another early object recognition system was ACRONYM by Brooks [27]. It is the first system designed to operate on noisy and incomplete image representations as it meant to be a general vision system. Its limitations were weak segmentation and limited interpretation [16]. Image prediction and matching were not sufficiently general for scenes with many objects.

The *Generalized Hough Transform* [18] is analyzed as a method for recognizing objects from noisy data in complex cluttered environment in the work of Grimson and Huttenlocher [39]. It was shown that the Hough transform should be adequate for the recognition of objects with limited occlusion and moderate sensor uncertainty, using isolated points such as vertices as the matching features. This method scales poorly when applied to complex, cluttered schemes, or when using extended features, such as edges, which are subject to partial occlusion [18]. In these cases however, the generalized Hough transform may still be useful for identifying matches that will be verified further.

Huttenlocher and Ullman [68] researched the *Alignment* method and developed the object recognition system ORA. They showed that the correspondence of three noncollinear points is sufficient to determine the position, 3D orientation, and scale of a rigid solid objects with respect to a 2D image.

Moreover, Basri developed in [20] a method that combines alignment with indexing and performs recognition by prototypes. The author used categorization as an indexing tool. The objects are divided into classes, where a class contains objects that share a fair number of similar features. Categorization is done by aligning the image to individual models of its class.

Another approach for alignment was developed by Ullman and Basri [146] for recognition by linear combination of models. The modeling of objects is based on the fact that for many continuous transformations of interest in recognition, such as rotation, translation, and scaling, all the possible views of the transforming objects can be expressed as the linear combination of other views of the same object. They proved that in the case of an object with sharp edges, two views are sufficient to determine the objects structure within an affine transformation and three were required to recover the full 3D structure of a rigidly moving object. For objects with smooth boundaries, three images were required to represent rotations around a fixed axis and five images were required for general rotations in 3D space.

Another method developed by Belongie *et al.* [22]. The method measures similarity between shapes and exploits it for object recognition. The approach has three stages: (1) solve the correspondence problem between two shapes using the *Shape context* descriptor, (2) use the correspondences to estimate an align transform, and (3) compute the distance between the two shapes as a sum of matching errors between corresponding points, together with a term measuring the magnitude of the aligning transformation. Figure 3.1) gives an overview of the computation Shape context features. Recognition is then treated in a nearest-neighbor classifier framework. The advantage of this method is that it can be used for a variety of shapes, such as silhouettes, trademarks, handwritten digits, and 3D objects.

Lowe developed in [90] an object recognition system, namely SCERPO, based on *perceptual organization* [89]. The system recognizes known 3D objects in single grayscale images, where objects are modeled as polyhedral and grouping is made on the basis of proximity, parallelism and co-linearity of the edges. Figure 3.2 gives an overview of the main processes of the vision system.

The system introduced by Havaldar and Medioni [57] dealt with noise and occlusion, as well as be able to do generic recognition using a perceptual grouping hierarchy. Groups were based on the proximity, parallelism, parallel and skewed symmetry and

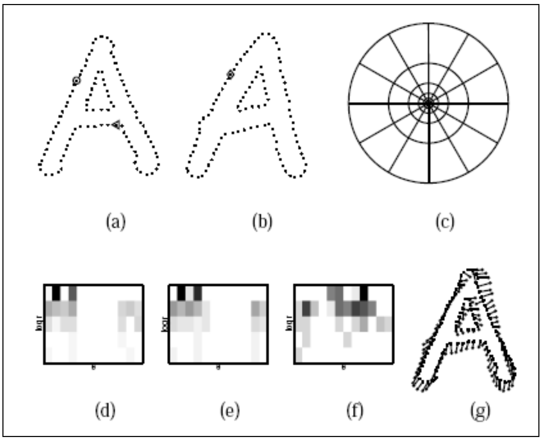


Figure 3.1: *Shape context computation and matching [22].* (a,b) Sampled edge points of two shapes. (c) Diagram of log-polar histogram bins used in computing the shape context. (d-f) Example shape context for reference samples. (g) Correspondence found using bipartite matching.

closure. Similar groups are grouped further into sets. The representation and matching of these sets is done using graphs. The system is able to handle generic recognition and occlusion.

3.1.2 Appearance-based Approaches

One approach to appearance-based object recognition was based on Principle Component Analysis (PCA). One of the first systems from this category was developed by Turk and Pentland [145] for face recognition. Mathematically, they were looking for the principle components of the distribution of faces, or the eigenvectors of the covariance matrix of the set of face images, treating an image as a point (or vector) in a very high dimensional space. Each eigenvector accounts for the variation among the face images.

The eigenface approach was generalized by Murase and Nayer [107] to handle complete parameterized models of objects. They represent each as a parametric manifold in two eigenspaces. The universal eigenspace [107] is computed using all objects of interest to the recognition system and is used for discriminating between objects. The

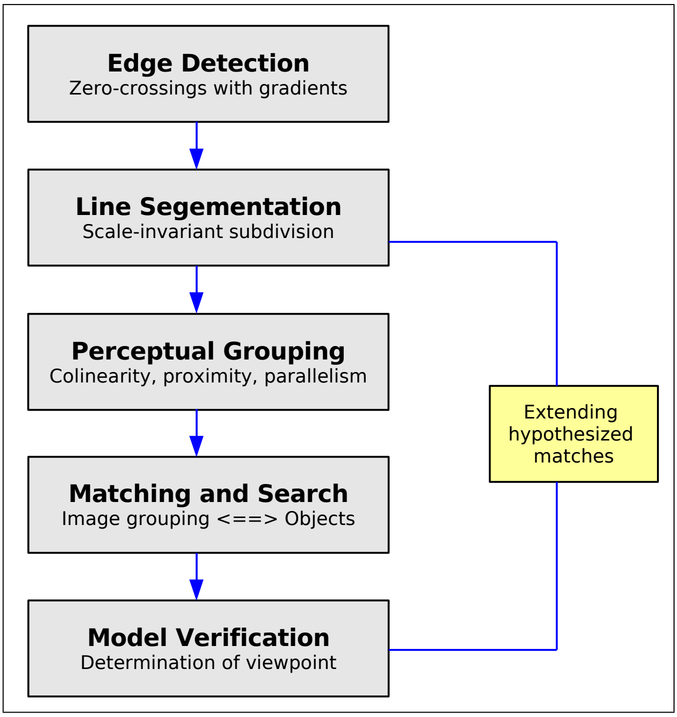

Edge Detection
Zero-crossings with gradients
Line Segementation
Scale-invariant subdivision
Perceptual Grouping
Colinearity, proximity, parallelism
Matching and Search
Image grouping <==> Objects
Model Verification
Determination of viewpoint
Extending hypothesized matches

Figure 3.2: The SCERPO vision system

object eigenspace is computed using only images of an object and is used for pose estimation. Recognition requires images to be normalized for size and 2D orientation.

The method introduced by Murase and Nayer [107] was further generalized by Hung *et al.* [161] into a system that is able to handle occlusion. Objects are decomposed into parts, where parts are polynomial surfaces approximating closed, non-overlapping image regions that optimally partition the image in a minimum description length sense. An object is completely characterized for different sensors and light sources, using the concept of *appearance of a apart*: two parts segmented from two images of the same object obtained with similar sensors and illumination configurations, are said to be appearance of the same parts if they are judged to have similar polynomial approximations in similar image locations.

An approach to solve the problems related to occlusion, cluttered background and outliers that eigenimage approaches usually have was proposed by Leonardis and Bischof [85]. The novelty of their approach lies in the way the coefficients of eigenimages are determined. Instead of computing the coefficients by projecting the data onto the eigenimages, they apply random sampling and robust estimation to generate hypotheses for the model coefficient. Computing hypotheses is then done as a selection procedure based on the Minimum Description Length principle.

Chen and Stockman [31] took an appearance-based approach to indexing. A 3D object is modeled by a collection of 2 $1/2$D views called *model aspects* made from 5 images taken by rotating the viewpoint up, down, left and right of a central viewpoint. The silhouette of the central edge map is extracted by mapping this map against the other four edge maps and then segmented into codons, which are formed by segmenting curves at minima of curvature. During recognition, the invariant features are extracted using a part segmentation algorithm. These features are used for indexing into a hash table to select hypotheses which are ordered using different voting schemes, such as majority voting. The final result is obtained through a verification step.

An appearance-based alignment approach was presented by Huttenlocher and Lorigo [67]. The method decides whether or not two planar point sets are views of the same 3D point set under orthographic projection, and constraints the 3D set up to an affine transformation of space. The advantage of this method is that it does not require any prior knowledge of the 3D structure of that object.

A method called probabilistic alignment was developed by Pope and Lowe [121]. In their method, an image is represented by a graph with nodes denoting features (such as edges, groups of edges, regions of uniform color etc.) and arcs denoting abstraction and composition relations among them. An object is modeled by a series of model views which is represented by a graph similar to an image graph. The model view is different from an image graph that describes, for each model feature, a distribution of where that feature may be expected to be found once the model and image have been satisfactorily aligned by a transformation. A match is a consistent set of pairings between some model and image features, in addition to a transformation closely aligning paired features.

Schmid and Mohr [134] developed an appearance-based system that can identify and locate objects in the case of partial visibility, image transformations and complex scenes. The approach is based on the combination of differential invariants computed at key points with a robust voting algorithm and semi-local constraints.

Abd-Al-Wahab *et al.* [12] presented a model for 3D object recognition from 2D views using Hu moment invariants [65] with SVM. Moreover, they combined the Hu invariants with Affine moment invariants [47] in [10]. In [11], they used simple color features with SVM for recognition and achieved good performance.

3.2 Generic Object Recognition

Now, we move the discussion to the more difficult problem of GOR by reviewing the body of literature in this problem. According to our interest, we divide the approaches into four categories:

1. Appearance-based approaches.
2. Shape based approaches.
3. 3D recognition approaches.
4. Multi-class recognition approaches.

3.2.1 Appearance-Based Approaches

Agarwal and Roth [13] first came up with the idea of a ”codebook” as a collection of class specific patches (see figure 3.3). The images in their approach are represented by binary feature vectors, which encode which image patches from the codebook are found in an image. learning is performed using a Sparse-Network-of-Winnow (SNoW). They used the UIUC dataset (see section 2.4) for evaluating their approach.

The idea of the ”codebook” representation was used by different approaches such as Csurka *et al.* [34]. They presented the ”Bag of Keypoints” approach, which is a straight forward but powerful one. The main steps of their approach are: 1) image patches are detected within the images and then described using a suitable descriptor 2) patch descriptors are then assigned to a set of predetermined clusters (a vocabulary) with a vector quantization algorithm, 3) then, a bag of keypoints is constructed. It counts the number of patches assigned to each cluster, and 4) a SVM classifier if finally applied, treating the bag of keypoints as a feature vector and thus determine which category(s) to assign to the image. They evaluated their approach on the Caltech dataset (see section 2.4) and achieved good performance.

Inspired by the ”codebook” idea presented by Agarwal and Roth [13], Leibe *et al.* [80]

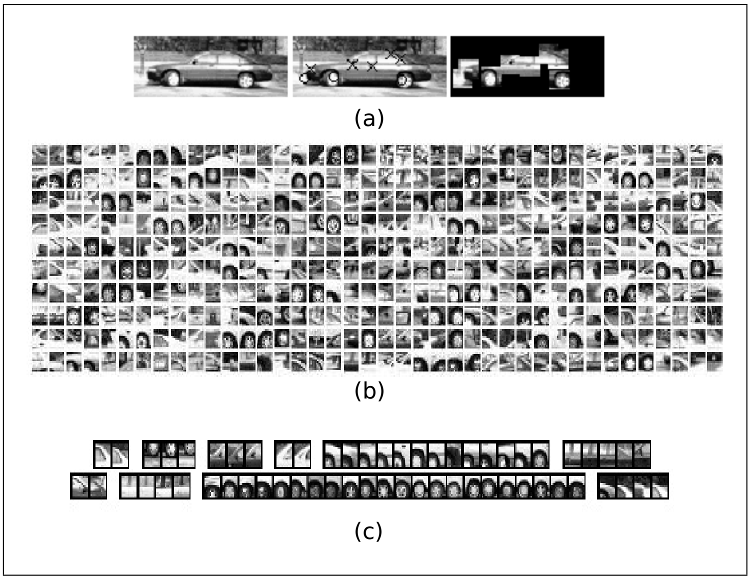


Figure 3.3: *The codebook representation [13].* (a) Detection and extraction of interest patches from images. (b) The vocabulary of parts extracted from images. (c) Example of the clusters formed after grouping the parts (patches) together.

presented an approach for object recognition and detection. The scheme first finds a set of regions for each training image, then clusters them in the manner in [13]. Additionally, for each cluster, the relative location of the object center and the average foreground/background mask is recorded. In recognition, interest points are again found and then a probabilistic Hough scheme is used to vote for the position of the object within the image, based on the match of the regions to each of the clusters. The maximum in voting space is found and used to project back into the image the regions which belong to the object. Then the foreground/background masks of each cluster can be used to provide a segmentation of the object. A large degree of supervision is required in training since each training example must be manually segmented.

Fritz *et al.* [52] added a discriminative second stage (SVM) to the model of Leibe *et al.* [80], where the performance is improved with the combination of the generative and discriminative techniques. Figure 3.4 shows the main stages of the model.

Moreover, Jurie and Triggs [72] showed how the clustering of codebook entries can be

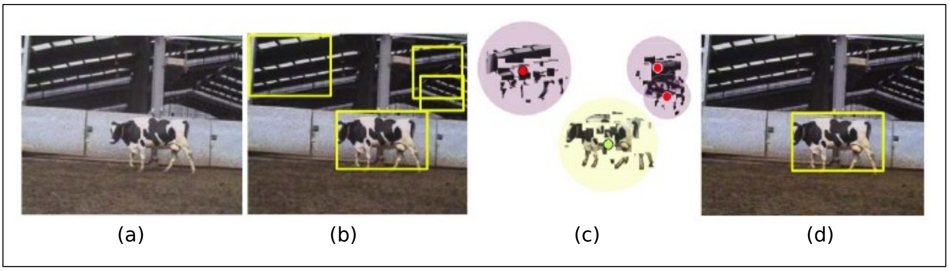


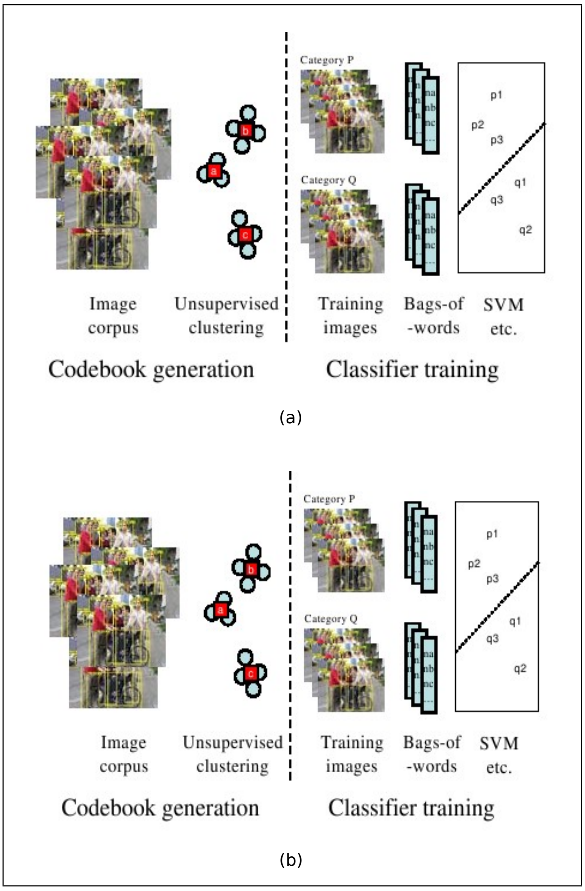
Figure 3.4: *Stages of the model of Fritz et al. [52].* (a) Original images. (b) Generative part: hypotheses detected by the representative Implicit Shape Model (ISM) [80]. (c) discriminative part: input to SVM stage. (d) Verified hypotheses.

done with respect to the non-uniform statistics of image patches.

Winn *et al.* [158] also used codebook representation in their recognition model, where a dictionary of local patches of textures is generated. They created a compact dictionary of words represented by Gaussian Mixture Models (GMMs).

Recently, Yang *et al.* [160] proposed a framework which unifies codebook generation with classifier training. Moreover, they express the partial similarity between features by constructing a category-specific visual words for each feature rather than quantizing the features using a single codebook. Figure 3.5 displays a diagram which shows the difference between their new approach and the standard approaches.

Another object recognition approach was introduced by Dorko and Schmid [37]. In training their model, regions are extracted from training images and clustered using EM. For each cluster, a score is computed, measuring its ability to discriminate between the foreground and the background classes. The top few clusters based on their discrimination scores are then selected to form a final classifier. In recognition, regions are extracted from the query image and assigned to the selected cluster or to the remaining background ones. A simple threshold on the number assigned to the selected clusters is used to perform classification. Good results are achieved on the Caltech datasets. The approach makes use of a hybrid generative/discriminative scheme in learning: the clustering of commonly occurring features, followed by a discriminative procedure to find the clusters that are distinctive of the class. The later stage is important since low level features like corners and edges occur very often but carry little information about the class label.



(a)

(b)

Figure 3.5: *Differences between standard codebook approaches and the approach of Yang et al. [160].* (a) Standard ”codebook” approach where the visual codebook generation and classifier training are separated. (b) Proposed approaches by Yang *et al.* [160] where the two phases are interleaved into a single optimization framework and the representation and classifications are iteratively refined [160]. Image is from [160].

A different approach to GOR is presented by Fergus *et al.* [43]. They used the constellation model proposed by Leung *et al.* [86] and the EM type of weber *et al.* [157]. In their work, Fergus *et al.* presented a method to learn and recognize object class models from unlabeled and unsegmented cluttered scenes in a scale invariant manner.

Objects are modeled as flexible constellations of parts. A probabilistic representation is used for all aspects of the object: shape, appearance, occlusion and relative scale. An entropy based feature detector is used to select regions and their scale within the image. In learning, the parameters of the scale-invariant object model are estimated using expectation-maximization in a maximum-likelihood setting. In recognition, this model is used in a Bayesian manner to classify images. Figure 3.6 gives an example to the recognition model. The flexible nature of the model is demonstrated by excellent results over a range of datasets including geometrically constrained classes (e.g. faces, cars) and flexible objects (such as animals).

Fergus *et al.* extended the constellation model in [44] to include heterogeneous parts consisting of curve segments and appearance patches.

In [45], Fergus *et al.* introduced a heterogeneous star model which reduces the learning complexity of the constellation model

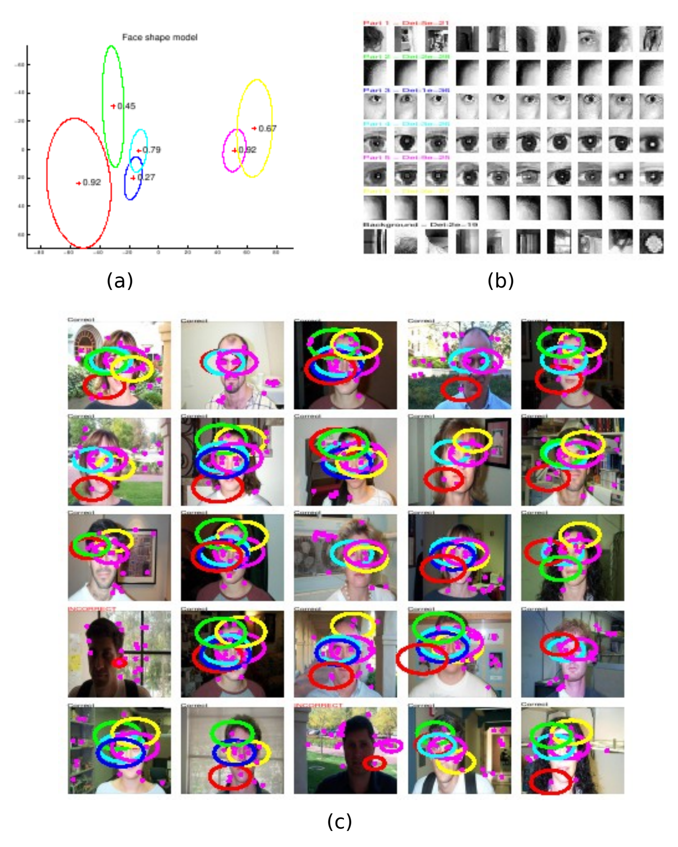
Caputo *et al.* [29] used SVM with local features via a new class of Mercer kernels in their recognition model. By this class of kernels, they perform scalar products on feature vectors which consist of local descriptors around interest points. They evaluated their model using the ETH80 and Caltech datasets.

Viola and Johns [151] used Boosting as the learning technique for their fast face detector. The images are represented using the integral image representation. Boosting selects afterwards a small number of visual features from a large computed set in a ”cascaded” manner, which allows the background regions of the images to be quickly discarded while spending the computation time for the object-like regions (see figure 3.7). The weak hypothesis they used is a thresholded average brightness of collections up to four rectangular.

In [153], Viola *et al.* extended their approach by incorporating motion information, where the approach is trained on manually pre-segmented data.

Opelt *et al.* [113, 111] used Boosting as the underlaying learning technique for their object recognition approach. They tried different combinations of point detectors and descriptors based only on appearance information. They evaluated their model using the Caltech dataset as well as their Graz01 and Graz02 datasets.

Zahng *et al.* [162] used also Boosting in their model. They presented an object class recognition approach which combines local texture features (PCA-SIFT), global features (shape context) and spatial features within a single multi-layer AdaBoost model of object class recognition. A two-layer AdaBoost training network is used. Figure



(a) (b)

(c)

Figure 3.6: *A face model (with 6 parts) developed by [43].* (a) Shows the shape model where the ellipses represent the variance of the parts and numbers represent the probability of each part being present. (b) 10 patches closest to the mean of the appearance density for each part and the background density. (c) Some sample images with a mix of correct and incorrect classifications.

3.8 displays a semantic overview for the recognition model. The function of the first layer is to choose the set of the local (PCA-SIFT) and global (shape context) features that best describe the object class. These two set of features are then boosted into a strong layer 1 classifier. Layer 2 boosting requires first to locate the good features from each sample based on the distances between the most discriminative local features se-

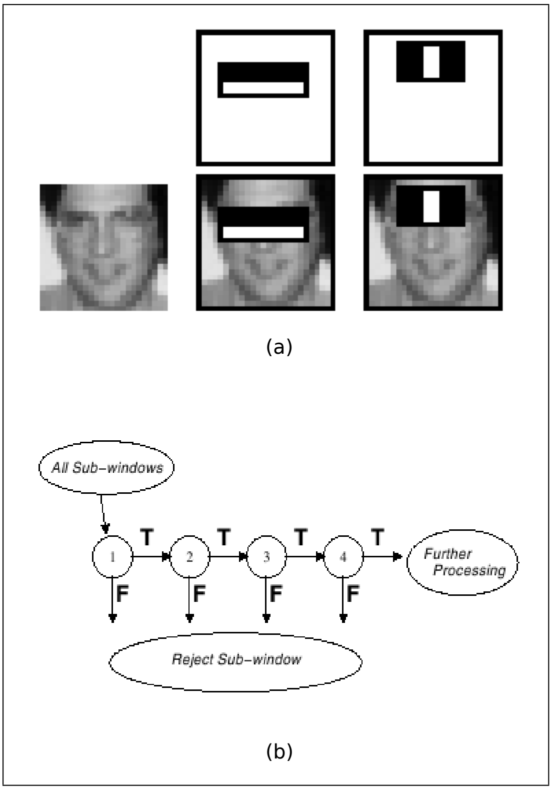


Figure 3.7: *Recognition model of Viola and Johns [151].* (a) Example feature representation. (b) Schematic illustration of the cascade detection model.

lected by layer 1. Pairwise spatial relationships (PSR) are then computed between these features using the method described in [19]. These PSR features are then given to the second layer of AdaBoost. An image is classified as containing an object class if conditions set on both classifiers are satisfied. They used the Caltech and GRAZ01 databases in their experiments.

Another approach was introduced by Thuresons and Carlsson in [143], which is based on histograms of qualitative shape indexes. These indexes are computed from combinations of triplets of locations and gradient directions in the images. The object categories are then represented by a set of histogram representation of training images. When a new image is presented, the inner products of the histograms of this image

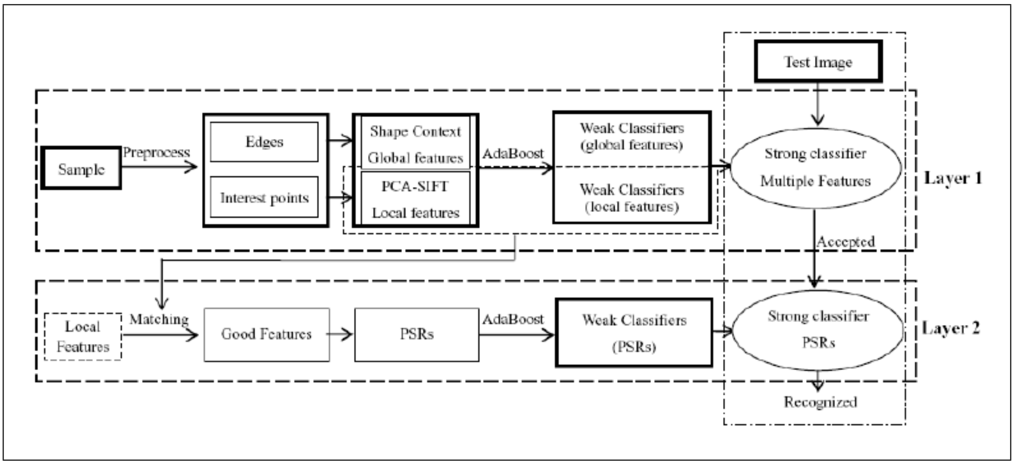


Figure 3.8: An overview of the recognition model of Zahng *et al.* [162].

with all training images histograms are calculated. The smallest of these products and the thresholds are used to recognize this image. The objects are required to be manually pre-segmented to reduce the representation and recognition complexity. Caltech dataset is used in their evaluations.

Moosmann *et al.* [104] presented an approach for GOR using visual attention. They proposed a classifier that combines saliency maps with an object part classifier. Prior knowledge stored by the classifier is used to simultaneously build a map online as well as to provide information about the object class. They evaluated their approach on several datasets including Graz02 dataset.

In [105], Moosmann *et al.* recently introduced a new clustering scheme, called Extremely Randomized Clustering Forest (ERCF) and used it for vector quantization for visual information. They combined the ERCF and the visual saliency maps as in [104] for visual information representation. SVM is used afterwards for classification as shown in figure 3.9.

Recently, Mutch and Lowe [108] presented a biologically-based model for GOR based on the model of Serre *et al.* [135]. In their model, images are reduced to feature vectors, which are classified afterwards using SVM. However, the features are computed hierarchically in five layers: an initial image layer and four subsequent layers, each is

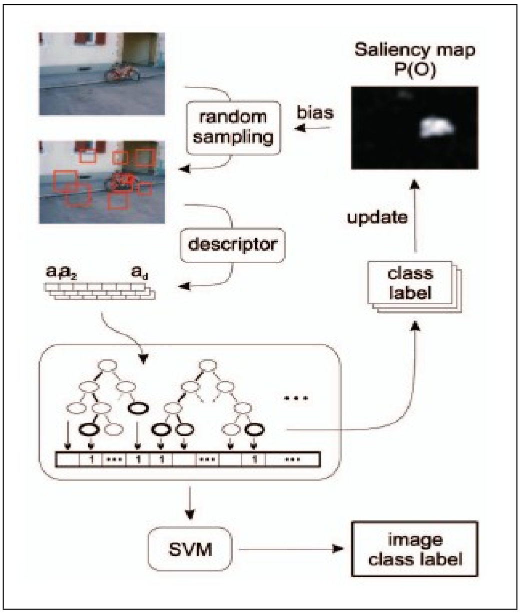


Figure 3.9: The classification model of Moosmann *et al.* [105] using ERCF and adaptive saliency map.

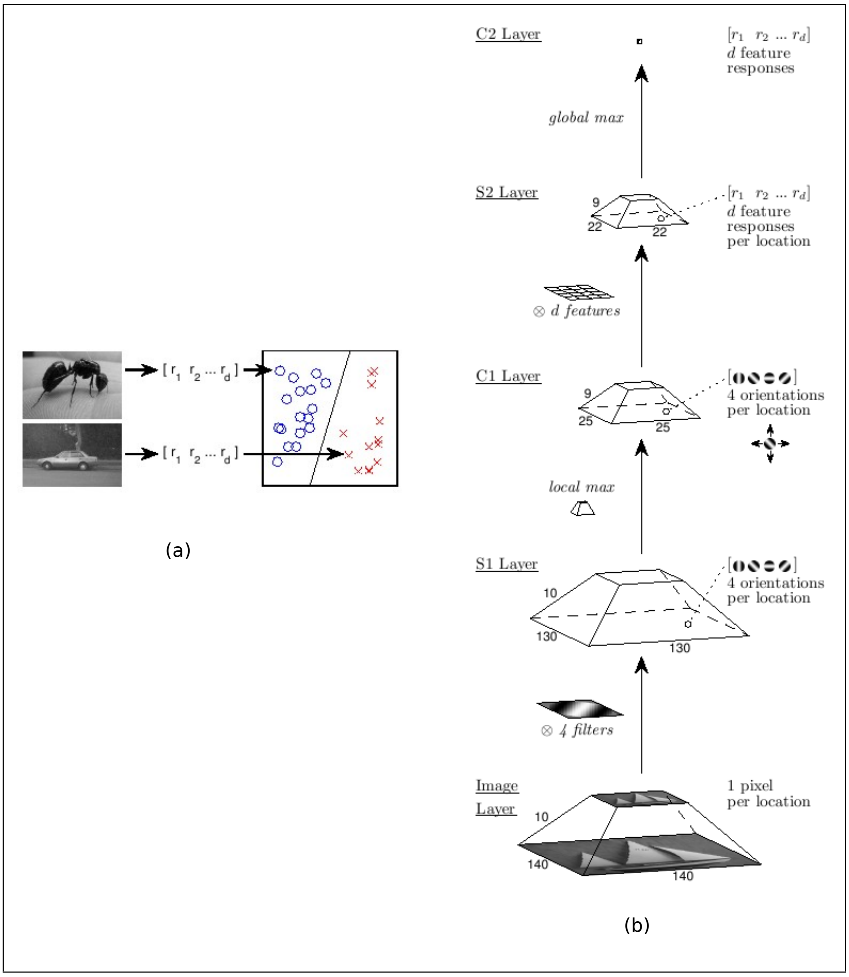
built from the previous using ”H-Max” model as shown in figure 3.10.
Thomas *et al.* [142] integrated the multi-view specific object recognition model proposed by Ferrari *et al.* [46] and the Implicit Shape model of Leibe and Schiele [83] in a new GOR model. The new model is able to recognize new object instances from arbitrary viewpoints.

All previous approaches used different degrees of supervision (*i.e.* at least weak supervision). The following approaches used almost no supervision in learning.

Sivic *et al.* [136] introduced an approach using probabilistic Latent Semantic Analysis (pLSA) from text analysis and applied it to images as collections of visual words.

Hillel *et al.* [64] presented an approach that learns a generative appearance model in a discriminative manner. Boosting is used to learn a decision boundary in feature space.

Kushal *et al.* [78] proposed a novel framework for GOR where object classes are



(a)

(b)

Figure 3.10: *The biologically-based GOR model of Mutch and Lowe [108].* (a) Overall model where images are reduced to feature vectors classified by SVM. (b) Feature computation in the model.

represented by assemblies of Partial Surface Models (PSMs). These PSMs are formed in a dense locally rigid assemblies of image features.

3.2.2 Shape-Based Approaches

Opelt *et al.* [116] used shape features for GOR and detection. They introduced a Boundary-Fragment Model (BFM) based on the work of Leibe *et al.* [80]. With their BFM, they capture the global geometry of the object category by capturing information about internal and external object boundaries. Boosting is used to select discriminative combinations of boundary fragments (weak detectors) to form a strong BFM detector.

Berg *et al.* [24] presented a model that depends on deformable shape matching using correspondence finding algorithm. Their algorithm is an integer quadratic program, where the cost function is a combination of geometric blur descriptors and geometric distortion between feature points. A nearest neighbot classifier is used for recognition.

Felzenszwalb and Huttenlocher [42] presented a ”Pictorial model”, with which features in the image are matched to parts of the model using a parts and structure model. The model is used to detect people in images (see figure 3.11)

Kumer *et al.* [77] extended the method of Felzenszwalb and Huttenlocher [42] into a probabilistic way to complete graphs. They used parts of outlines in their application of pictorial structure.

Leibe *et al.* [84] improved the method in [80] by including shape information to detect pedestrians. They used a verification step that uses Chamfer matching of a representation of the whole contour of the image.

Fergus *et al.* [44] extended their constellation model by using boundary curves between bitangent points.

Amores *et al.* [15] proposed a recognition model where they learn contextual information in the form of generalized correlograms as class representations using Boosting. In their feature, they combine the local information (in the form of structure and color) with spatial relations using correlograms that encode the edge locations around a point in a log-polar quantization.

3.2.3 3D Recognition Approaches

A small number of research have investigated the problem of generic 3D object recognition due to its difficulty. One of these approaches is presented by Savarese and Fei-Fei [130]. In their approach, a model of an object category is captured by linking together diagnostic parts of objects from different viewing points. These parts are large and discriminative regions of the objects and consists of many local invariant features.

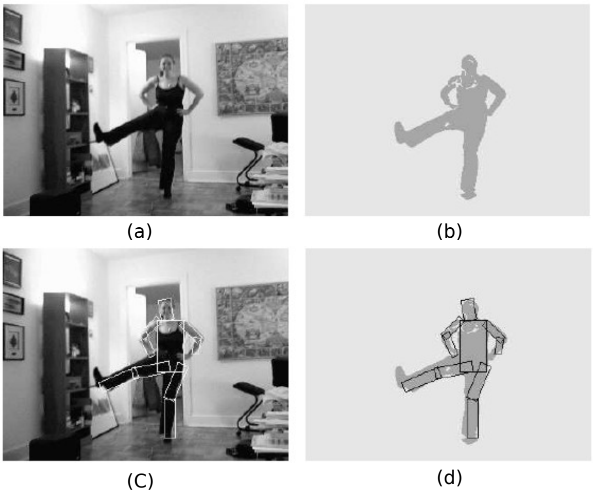


Figure 3.11: *An example of input and matching results using the Pictorial structure of Felzenszwalb and Huttenlocher [42]*.(a) Input image. (b) Binary image obtained by background subtraction and used for finding a configuration that covers the object of interest. (c)-(d) Match results superimposed on both images respectively.

To form a model of the object class, the parts are connected through their mutual homographic transformation. The resulting model is a summarization of both appearance and geometry information of the object class. Figure 3.12 gives an overview of the main ideas of the model. Additionally, the authors introduced a new object category dataset for 3D object recognition tasks (see section 2.4).

Savarese and Fei-Fei [131] extended their recognition framework in [130] and improved it to be able to recognize previously unseen poses based on the works in single object view synthesis.

Recently, Sun *et al.* [138] proposed a generative probabilistic framework for learning visual 3D object categories. In their model, an object is represented as a coherent ensemble of parts linked across different viewpoints where each part is represented by a distribution of appearance elements. Then, a generative model is used for learning the relative position of parts within each viewpoint and also the corresponding parts locations across viewpoints.

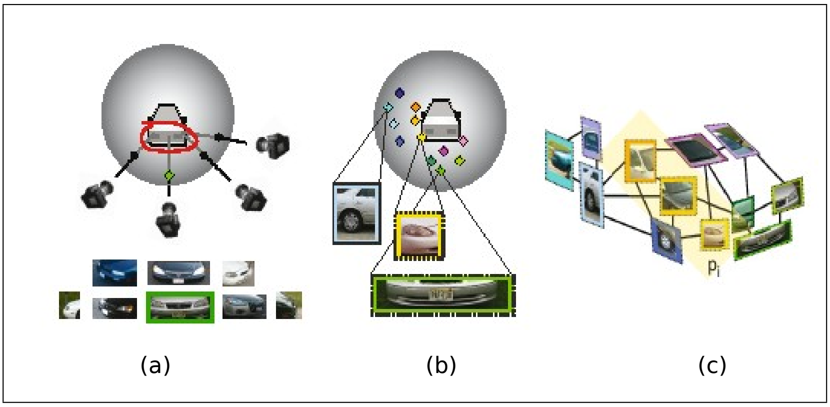


Figure 3.12: *The main idea of the model of Savarese and Fei-Fei [130].* (a) Canonical parts of a car. (b) Location of canonical parts. (c) Canonical parts are connected together in a linkage structure.

Su *et al.* [137] extended recently the framework of Sun *et al.* [138] to be able to recognize unseen views by pose estimation and synthesis using a dense, multiview representation of the viewing sphere parameterized by a triangle mesh of viewpoints. Moreover, they proposed an incremental learning algorithm to train the generative model proposed in [138].

A different approach is described by Ruiz-Correa *et al.* in [129]. The approach developed to recognize objects belonging to a particular shape class in range images. In their approach, first, shape class components are learned and extracted from range images. Then, the spatial relationships among the extracted components are encoded using a shape representation called symbolic surface signature. This results in forming a shape class model that consists of a three-level hierarchy of classifiers, where the first two levels of the hierarchy extract the component and the third one verifies their geometric relationships. Figure 3.13 displays a recognition example of the model. The dataset used for the purpose of learning and classifying the model is a collection of range images of objects made of clay. The dataset is then enlarged by applying deformations to the original clay objects to offer intra-class variabilities (see figure 3.13).

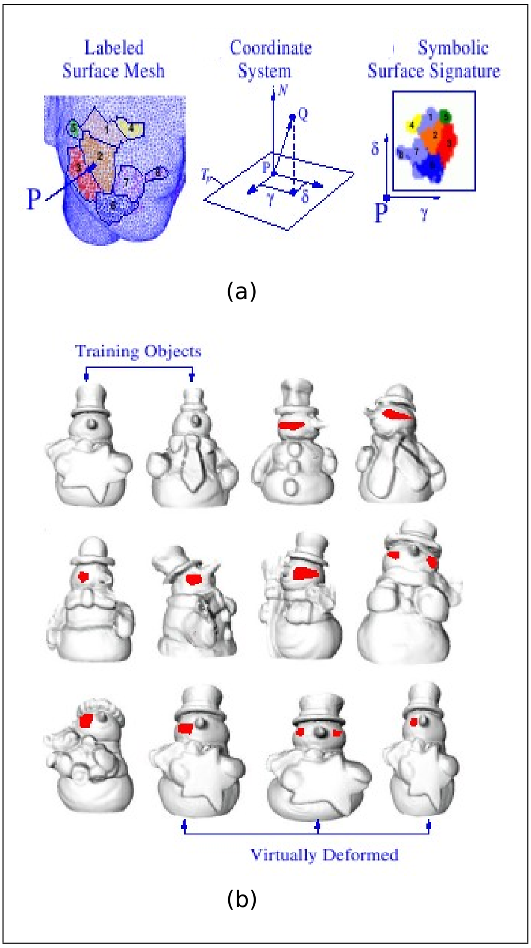


Figure 3.13: *Recognition model for shape classes from range images proposed by Ruiz-Correa et al. in [129].* (a) The symbolic surface signature for a point P on a labled surface mesh model of a human head [129]. (b) Recognition examples using the proposed recognition model.

3.2.4 Multi-Class Recognition Approaches

In specific object recognition, the developed recognition models can handle many different number of objects. However, in GOR, the recognition of many classes is still restricted.

Torralba *et al.* [144] presented the Joint Boosting algorithm for multi-class learning and recognition of object classes. The algorithm is based on the GentleBoost algorithm and the authors build a strong classifier using regression stumps to be shared among the different classes. Using their model, 21 different object categories are jointly learned.

Opelt *et al.* [115] extended their Boundary-Fragment model for multi-class learning. Based on the Joint Boosting algorithm, they presented a model where objects are learned jointly. Prior knowledge is used to learn new object classes in an incremental manner.

Fei-Fei *et al.* [41] used also prior information on the spatial distribution to help learning new classes.

Also, the models mentioned in [130, 131, 138, 137] perform multi-class recognition as well.

3.3 Conclusions

In this chapter, an overview of the body of literature in object recognition problem has been presented. First, the approaches developed for specific object recognition have been reviewed as specific object recognition is considered to be the basis for GOR. Then, early, recent and state-of-the-art GOR approaches have been presented. We can notice that the use of local appearance-based of information is the dominant among the different presented GOR approaches. Most of the developed models make use of local appearance information, such as texture, for recognition. Some approaches use additional information in the form of spatial relations among local features to improve the recognition performance. The use of completely shape or geometrical models with the aid of appearance is not addressed frequently in GOR.

Moreover, few approaches tackled the problem of generic 3D object recognition due to its difficulty. The approaches that addressed the problem used also appearance based information, while the use of range and 3D information is been ignored or neglected. One reason for this could be the lack of datasets which provide 3D information about its members and at the same time suitable for the GOR task.

We can also notice that most of the approaches performed binary instead of multi-class recognition. This could be argued to many reasons. One of these reasons is that binary classification is more easier than the multi-class case although the second is more realistic.

Chapter 4

Generic Recognition of 2D Objects

This chapter introduces our model for generic 2D object recognition using a combination of different local appearance cues. An overview of the whole model as well as an explanation of each of its different phases are presented. Experimental evaluation of the recognition performance of the model using different famous benchmarks is accomplished. Furthermore, comparisons of different boosting algorithms in the context of generic object recognition (GOR) are presented.

4.1 Motivation

In our approach, we perform local description of images without making use of any spatial relationships among the detected local regions. Only appearance-based information is used to describe these regions. We use the least amount of supervision we can by using only the labels of the training images as the only information given to the model about the different object classes during the process of learning (weakly supervised learning).

Generally, in generic object recognition, the performance of local descriptors (features) varies from a class category to another. One descriptor might have a good performance on one category and low performance on another. Combining more than one descriptor in recognition can give a solution to this problem. However, the choice of descriptor's type and number of descriptors to be used is, in this case, an important issue.

The idea of combining different information cues for recognition is not new. Many different approaches applied this idea in their recognition models such as [162, 113, 111]. Most of these approaches exploited a combination of different features and descriptors without beforehand investigating the suitability of their combination for the task to be accomplished. For example, Opelt *et al.* [114] proposed a GOR model using Boosting. They combined different three interest point detectors together with four types of local descriptors. They did not mention why they chose these features and why these features could be suitable for their recognition task. Also, a careful choice of features is supported by pattern recognition theorems such as Watanabe's *ugly duckling theorem* [156] since it is possible that two arbitrary patterns are made *similar* by encoding them with a sufficiently large number of redundant features.

Moreover, most of the GOR approaches such as [13, 43, 113, 111, 162] did not make use of any color information for recognition, although color could be a helpful aid in accomplishing this task.

In our model [58, 59], a careful choice of the descriptors to be combined is performed. Our idea is that, grayscale (texture) and color information are two important cues for objects appearance. Texture and color provide many useful information to characterize the objects and hence to differentiate among different objects classes. Therefore, we decided to use a combination of only two different types of descriptors: grayscale and color descriptors. This helps us to avoid the game of trail and error of combining many features and investigating their performance on the used classes as done, for example, in [114].

The major contributions presented in this chapter are as follows: 1) the presentation of an approach for 2D generic object recognition from grayscale and color features combination using weak supervision, 2) Comparing the performance of proposed different texture descriptors in the context of generic object recognition and 3) comparing boosting algorithms in the context of GOR.

Section 4.2 presents the proposed recognition model with an explanation of its different phases. Experimental evaluations of the recognition model using different recognition datasets are given in section 4.3. Section 4.4 presents performance comparisons of two texture descriptors in the context of GOR. Comparisons of different boosting algorithms in the context of GOR are given and evaluated in section 4.5.

4.2 The Recognition Model

4.2.1 Overview

In our generic recognition model objects from a certain class in still images are to be recognized. Objects are neither segmented before the learning process nor information about their location or position within the images are given in the learning. Figure 4.1 gives an overview of the general framework of our recognition model. In the first step, interest regions are detected in the training images. We exploit an affine invariant interest point detector [99] for accomplishing this task. Afterwards, local descriptors are extracted from the detected interest regions. A combination of two different types of local descriptors is used: grayscale (texture) descriptor and local color descriptor. In the next step of the model, local descriptors together with labels of the corresponding training images are given to a boosting learning algorithm [50] which produces a final classifier (strong hypotheses) as an output. The final classifier predicts if a relevant object is presented in the new (previously unseen) test image (*i.e.* binary classification). Figure 4.2 gives an example of the learning and recognition processes of our recognition model.

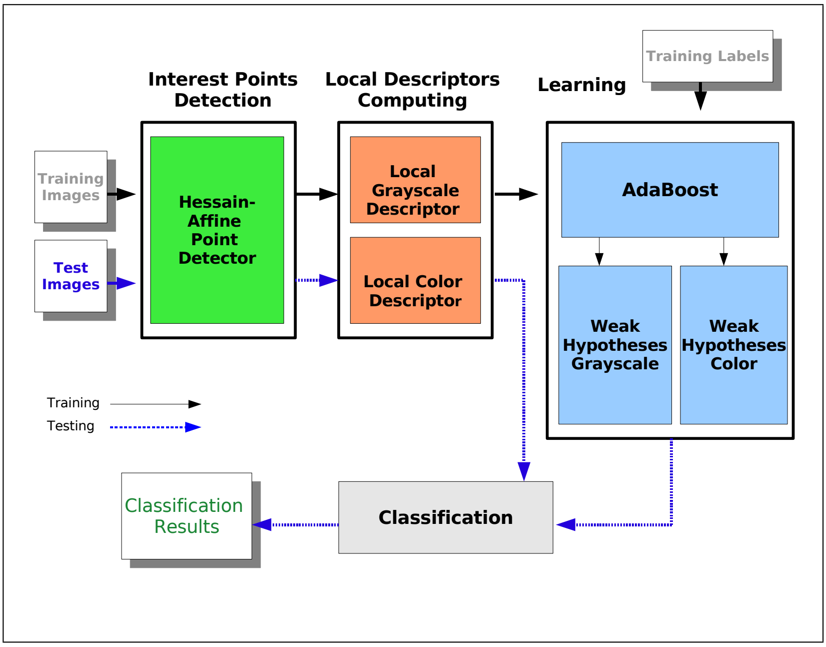


Figure 4.1: The general framework of our model for generic 2D object recognition.

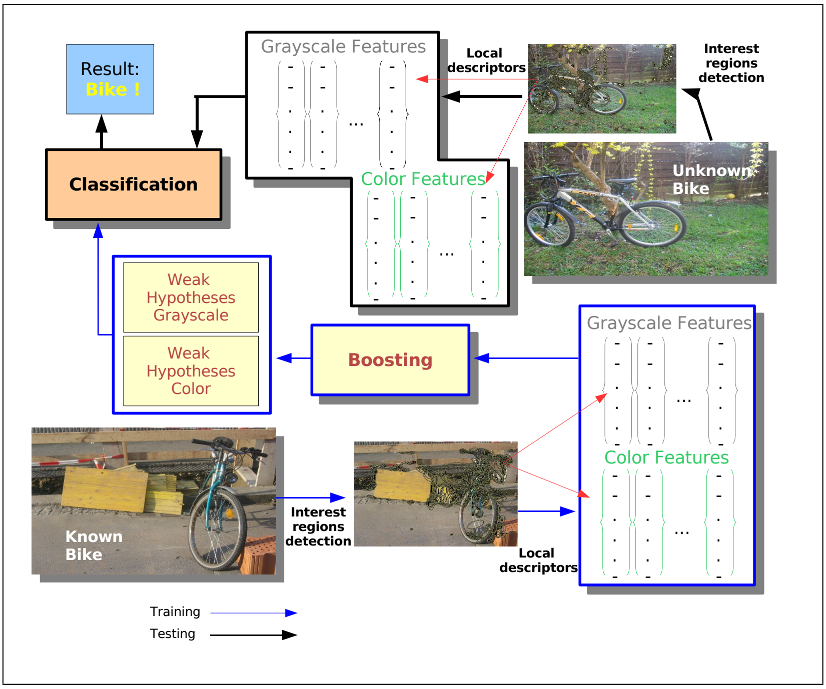


Figure 4.2: An example describing the different phases of our recognition model.

4.2.2 Interest Points Detection

There are different interest point detectors in the literature, each of which has some different properties. An overview of different interest point detectors is presented in section 2.1.2. Moreover, different evaluations of these detectors are also performed in order to measure and compare their performance in a certain context, mainly in the context of image matching and retrieval. Based on the evaluation of interest point detectors given by [106], we decided to use the Hessian-affine invariant detector [99, 100]. As this technique is state-of-the-art, we do not give details about it here. Interested readers are encouraged to refer to the given references for more details. We used the same parameter settings as the authors reported in their experiments.

4.2.3 Regions Description

As we previously mentioned, we use in our model a combination of two different descriptors which represent two different types of appearance information, namely texture and color. As a texture (grayscale) descriptor, we use the Gradient location-orientation Histogram (GLOH) [101], while we use the opponent color angle descriptor [148] for representing the color information. The reasons behind using these two descriptors are given next.

Gradient location-orientation Histogram (GLOH) descriptor [101] : is an extension to the SIFT descriptor [92], which is designed to increase its robustness and distinctiveness. To compute the GLOH descriptor, the SIFT descriptor is computed for a log-polar location grid with 3 bins in radial direction and 8 in angular direction, which results in 17 location bins (see figure 4.3(e)). The gradient orientations are quantized in 16 bins, which gives a histogram of 272 bins. The size of the descriptor is then reduced to 128 bins with PCA.
The superior performance of the GLOH descriptor over many other descriptors, as reported in [101], is the main reason of using it in our model. GLOH descriptors are computed with the code provided by [101].

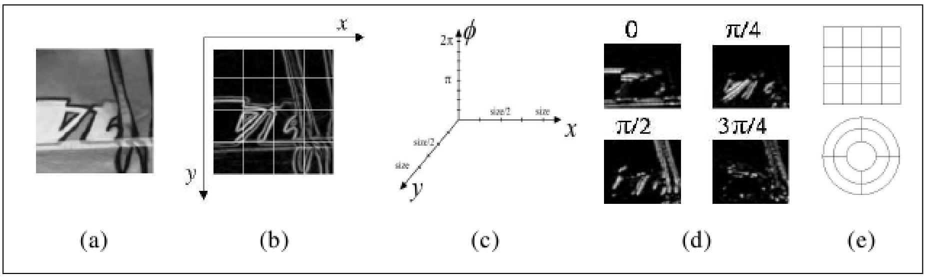


Figure 4.3: *SIFT descriptors*: (a) Detected regions. (b) Gradient image and location grid. (c) Dimensions of the histogram. (d) 4 of 8 orientations plan. (e) Cartesian and the log-polar grid. Image from [101]

Opponent color angle descriptors: Authors in [148] introduced a set of local color descriptors with different criteria such as photometric robustness, geometric robustness, photometric stability and generality. Among those descriptors introduced in [148], we chose to use the opponent angle color descriptor as it is robust with respect to both geometrical variations caused by changes in viewpoint, zoom and object orientations and photometric variations caused by shadows, shading and specularity. Brief description of how to construct the opponent angle color descriptor is given (according to [148]) as follows :

$$ang_x^O = \arctan(\frac{O1_x}{O2_x}) \tag{4.1}$$

where $O1_x$ and $O2_x$ are the derivatives of opponent colors and are given by:

$$O1_x = \frac{1}{\sqrt{2}}(R_x - G_x), O2_x = \frac{1}{\sqrt{6}}(R_x + G_x - 2B_x) \tag{4.2}$$

and R_x, G_x and B_x are the derivatives of color channels.
The opponent colors and their derivatives are proved to be invariant with respect to specular variations [148].
Before computing the previously mentioned invariant, color illumination normalization should be first done as described in [148]. They introduced two methods for normalization, zero-order and first-order normalization. We use the first-order color normalization which is recommend by [148] to be used with the opponent angle descriptor, and is given as:

$$Co^*(x) = \frac{Co(x)}{\overline{|Co_x(x)|}} \tag{4.3}$$

where the bar indicates a spatial average: $\bar{a} = \int_S adx / \int_S dx$, S is the surface of the patch, $Co \in \{R, G, B\}$ and Co_x is the derivative of the color channel.
To construct the opponent angle descriptor, the derived invariant is transformed into a robust local histogram. This is done by adjusting the weight of a color value in the histogram according to its certainty as in [148] (photometric stability). The resulting opponent angle descriptor is of dimension 37.

4.2.4 Learning Model

Boosting is the underlaying learning technique in our learning model. In the literature, there are many different boosting algorithms as mentioned in section 2.2.1. AdaBoost algorithm [132] is one of the most famous boosting algorithms. It generates a combined hypotheses with large margin and works well on data with low degree of noise. Our learning model is based on the AdaBoost algorithm. However, we use the AdaBoost version modified by [112] for GOR task (shown in algorithm 5). More information about the AdaBoost algorithm is given in section 2.2.1. Now, we will explain how learning the descriptors combination using the AdaBoost algorithm is accomplished.

An important part of any boosting algorithm is the weak learner. To adapt for a specific task, a suitable weak learner should be used. For the purpose of object recognition, we use a modified version of the weak learner presented in [112].

Since we use more than one descriptor to describe the images, each training image is represented by a set of features $\{F_{i,f}(t_{i,f}, v_{i,f}), f = 1...n_i\}$[1] where n_i is the number of features in image I_i, $t_{i,f}$ indicates the type of the feature (g for grayscale and co for color) and $v_{i,f}$ is the feature value. The AdaBoost algorithm (see algorithm 5) selects in each iteration, with the aid of the weak learner (see algorithm 6), two weak hypotheses: one for each different descriptor type. Each weak hypothesis consists of two components (selected and computed by the weak learner): a feature vector v_k^x and a certain threshold θ_k^x (a distance threshold) where $x = g$ for the grayscale descriptor and $x = co$ for color descriptor. The threshold θ_k^x measures if an image contains a descriptor $v_{i,j}$ that is similar to v_k^x. The similarity between $v_{i,j}$,which belongs to the image I_i, and v_k^x is measured using Euclidean distance for both descriptor types. Algorithm 6 gives an overview of our weak learner which is a modified version of the one described in [112]. The weak learner in [112] selects the best feature vector over the different descriptor types while in our modified version, the best vector in each descriptor type is selected.

In the recognition step, a test image I_{ts} is presented and a set affine interest points is detected. Grayscale and color descriptors are then extracted. For each weak hypothesis h_k^g and h_k^{co} and their associated feature values and thresholds, we find the grayscale and color features in the test image with the minimum distance $d(v_k^g, I_{ts})$ and $d(v_k^{co}, I_{ts})$ respectively. Then, we compare these minimum distances to the thresholds θ_k^g and θ_k^{co} respectively. A classification output for one weak hypothesis is computed as follows:

$$f_k^x = \begin{cases} +1 & \text{if } d\left(v_k^x, I_{ts}\right) < \theta_k^x \\ \\ -1 & \text{Otherwise.} \end{cases} \tag{4.4}$$

---

[1] The same mathematical notations that are used in section 2.2.1 are used overall the thesis.

This results in a classification output for each weak hypothesis $f_k^g, f_k^{co} \in \{-1, +1\}$. Then, the combination output is computed as follows:

$$h_k = \max(f_k^g, f_k^c) \tag{4.5}$$

After all hypotheses are processed, the output of the final (strong) classifier is compared to the threshold Ω (see AdaBoost algorithm 5) and the test image is then accepted or rejected depending on the output of the classifier. The final classifier threshold Ω is varied to get various points for the ROC curve. The threshold could be varied from ∞ to $-\infty$ and tracing a curve through ROC space could then be done. This method is used in many researches (e.g. [152]), but computationally, this is a poor way of generating a ROC curve as it is neither efficient nor practical [40]. In [40], an efficient algorithm for generating an ROC curve, which we use here, is presented and explained. It depends on exploiting the monotonicity of thresholded classification. Details about the method are found in [40].

4.3 Experimental Evaluation

We evaluate our recognition model using two different datasets, namely the Caltech 4 [2] and Graz02 [5] datasets. Although Caltech 4 dataset suffers from some limitations such as somewhat limited range of image variability [120], it played a key role, in addition to UIUC and Caltech 101 datasets, in the recent research of GOR during providing a common ground for algorithms development and evaluation. Caltech 4 has been used by many state-of-the-art GOR approaches for binary classification and Caltech 101 has been usually used by the approaches which perform multi-class GOR tasks (e.g [55]). We use Caltech 4 dataset in our evaluations to be able to compare the performance of our approach to state-of-the-art approaches. At the same time, we use Graz02 dataset, which is more recent and difficult one and avoids the problems exist in Caltech 4 dataset.

4.3.1 Experiments using Caltech 4 dataset

To compare our results to the existing approaches, we first use the Caltech 4 dataset in evaluating our recognition model. We use for training 100 images of the object class as positive examples and 100 images of the counter-class as negative examples. For testing, 50 positive examples and 50 negative examples are used. The features of each image are clustered to 100 cluster centers using the k-means clustering algorithm to

```
Input: Training images $(I_1, l_1), \dots (I_N, l_N)$.
Initialize: Set the weights distribution $w_1 = \cdots = w_N = 1$.
for $k = 1, ...T$: do
    (1) Train weak learner using distribution $\mathbf{w}_k$ and get a weak hypothesis $h_k$.
    (2) Calculate the classification error as:
        $\varepsilon_k = \frac{\sum_{i=1}^{N}(h_k(I_i) \neq l_i) w_i}{\sum_{i=1}^{N} w_i}$
    (3) Choose $\alpha_k = \sqrt{\frac{1 - \varepsilon_k}{\varepsilon_k}}$
    (4) Update the weights: $w_{k+1} = w_k \alpha^{-l_i h_k(I_i)}$ for $i$ = 1 to $N$
end
Output: Final hypothesis:
    $H_x = \begin{cases} +1 & \text{if } \sum_{k=1}^{T}(\ln \alpha_k) h_k(x) > \Omega \\ -1 & \text{Otherwise} \end{cases}$    (4.6)
.
```

Algorithm 5: AdaBoost algorithm. A modified version for generic object recognition task [112].

construct the images *signature* formed by the centers of its clusters and their relative sizes [119] .

The counter-class: The counter-class is important for training a classifier of a certain object class as it provides images for objects different from the one needed to be learned. We do not use the background class of the Caltech dataset as a counter-class (negative class), which is used by almost all the recognition approaches (that used the dataset) as the counter-class, because it does not contain colored images. Instead, we evaluate our recognition model using two different counter-classes: the background class of Graz01 dataset [4] and the leaves class of the Caltech dataset. Figure 4.4 shows some example images of each counter-class. These two different counter classes are more difficult than the background class of Caltech 4 as shown by the results in table

Input: Labeled representations $(R\left\{I_i\right\}, l_i)$, $i = 1, \ldots N$,
$R\left\{I_i\right\}\left\{(t_{i,f}, v_{i,f})\,|f = 1, \ldots, n_i)\right\}$.

(1)-Distance function: Let $d_t(.,.)$ be the distance with respect the description vectors of type t in the training images.

(2)-Minimal distance matrix: For all description vectors $(t_{i,f}, v_{i,f})$ and all images I_j calculate the Minimal distance between $v_{i,f}$ and description vectors in I_j,

$$d_{i,f,j} = \min_{1 \leq g \leq F_j : t_{i,g} = t_{i,f}} d_{t_{i,f}}(v_{i,f}, v_{j,g})$$

.

(3)- Sorting: For each i, f let $\pi_{i,f}(1), \ldots, \pi_{i,f}(m)$ be a permutation such that

$$d_{i,f,\pi_{i,f}(1)} \leq \cdots \leq d_{i,f,\pi_{i,f}(m)}$$

.

(4)- Select best weak hypothesis (Scanline): For all description vectors $(t_{i,f}, v_{i,f})$ of type t, calculate over all images I_j

$$\max_s \sum_{j=1}^{s} w_{\pi_{i,f}(j)} l_{\pi_{i,f}(j)}$$

and select the description vectors $(t_{i,f}, v_{i,f})$ where the maximum is achieved.

(5): Select threshold θ: With the position s where the scanline reached a maximum sum for each selected descriptor type t the, threshold θ is set to

$$\theta = \frac{d_{i,f,\pi_{i,f}(s)} + d_{i,f,\pi_{i,f}(s+1)}}{2}$$

.

Algorithm 6: Modified version of the weak learner of [112].

4.1. The results in table 4.1 are the output of recognition experiments we performed using Caltech 4 dataset with the same previously mentioned experimental settings. In

this experiment we use the GLOH descriptor (without combining it with the color features) with Hessian-affine point detector. We performed these experiments to show how difficult the counter-class (Graz01 background class and the leaves class) we use than that is been used (Caltech 4 background class) by almost all the approaches that used Caltech 4 dataset in their experiments. It is worth mentioning that each counter-class is used separately in the experiments. In other words, each experiment is repeated twice, each time using one counter-class.
The learning procedure are run for number of iterations $T = 100$ when using leaves as a counter-class and $T = 150$ when using the Graz01 background as a counter-class [2].

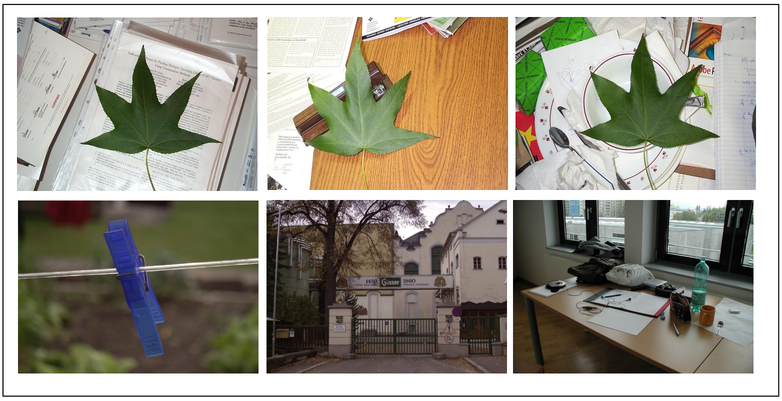


Figure 4.4: *Example images of the two classes used as counter-class in the Caltech 4 experiments*. Leaves class (first row) and Graz01 background class (second row).

First, we evaluated our model using each descriptor separately to be able to notice the recognition performance of each of them, and hence the benefits of combining them later. Afterwards, a combination of the two different descriptors is used. Table 4.2 displays the classification rates (true positive rates) at the ROC-Equal-Error rates of the recognition results using the Graz01 background class as a counter-class for the used four object classes while table 4.3 displays the results using the leaves class as the counter-class. Moreover, figure 4.5 displays the ROC curves of recognition results using GLOH-color (GC) combination with Graz01 counter-class (GC-G) and with leaves counter-class (GC-L) respectively. The results in both tables show the improvements

---

[2] These numbers were experimentally evaluated.

Table 4.1: Classification rates at ROC-eqq.-err. rates of our recognition results using the Caltech 4 dataset with different counter-classes (Caltech 4 background, Graz01 background and Leaves class).

Class	Using Caltech 4 background	Using Graz01 background	Using leaves class
Motor	100 %	92%	94%
Cars	100 %	80 %	82%
Airplanes	96 %	78%	78%
Faces	100%	92%	92%

we gain in performance from combining the two descriptors together. Each descriptor achieves a high performance on some classes and relatively good performance on others. Using a combination of the two descriptors improves the recognition performance on almost all the classes.

Table 4.4 displays a comparison of the recognition performance of our model to the results of state-of-the-art GOR approaches and models. The comparison reveals the good performance of our model using the two different counter-classes (GC-G and GC-L). Our model (GC-G) achieves the best results for the two classes ”motors” and ”cars” while our model (GC-L) yields the second best results for the class ”faces”. A reasonably good performance for the class ” airplanes” is however achieved. We can measure the overall performance of each approach as the average classification rate over the used classes (see last column in table 4.4). This measurement encapsulates the performance of each recognition model and reveals the good results achieved by ours (GC-G).

Note that the amount of supervision varies over the approaches presented in the table 4.4. The approach in [143] uses class labels and bounding boxes around the classes for training (high supervision) while the approaches in [15, 19, 29, 43, 114, 45] use only class labels for training (weak supervision like in our approach). The approach in [136] uses no supervision. Moreover, it should be noted that the counter-class used in our model is more difficult than the one used by all other approaches as shown in table 4.1.

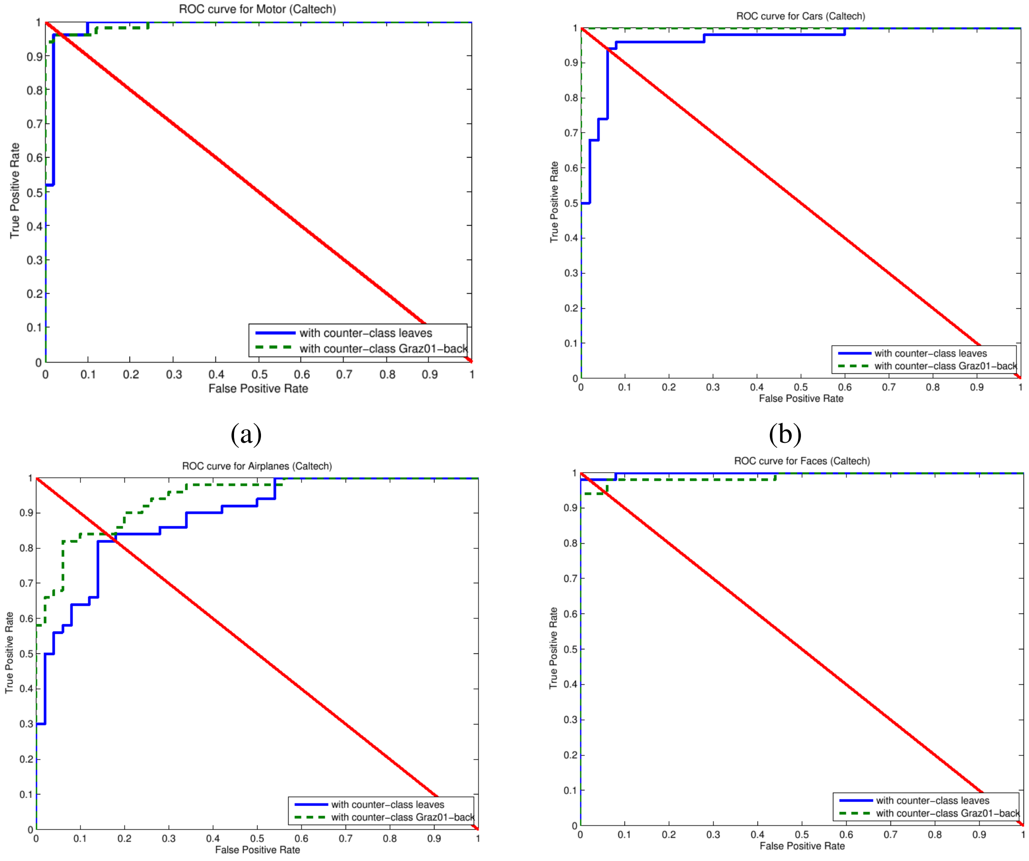
To give more insight on the recognition performance of our model, figure 4.6 gives examples of incorrectly classified images which are classified as false negatives (4.6 (a) or as false positives (4.6 (b))). However, the images in figure 4.6 are simple to be incorrectly classified while the dataset contains more difficult images which are correctly classified.

Table 4.2: Classification rates at ROC-eqq.-err. rates of our results using (GC) feature combination on the Caltech 4 dataset with using the Graz01 background class as the counter-class.

Class	GLOH	Opponent angle	Combination (GC-G)
Motors	92%	94%	**96**%
Cars	80%	100%	**100**%
Airplanes	78%	80%	**84**%
Faces	92%	86%	**94**%

Table 4.3: Classification rates at ROC-eqq.-err. rates of our results using (GC) feature combination on the Caltech 4 dataset with using the leaves class as the counter-class.

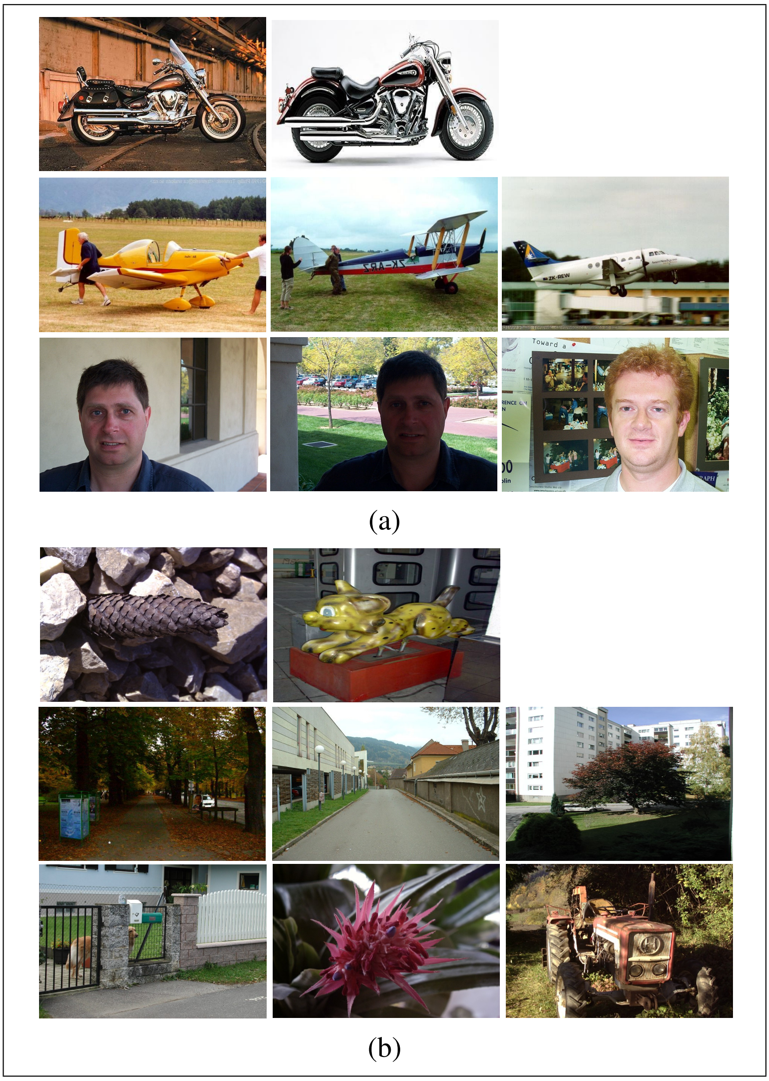
Class	GLOH	Opponent angle	Combination (GC-L)
Motors	94%	82%	**96**%
Cars	82%	86%	**94**%
Airplanes	72%	80%	**82**%
Faces	92%	94%	**98**%



(a) (b)

(c) (d)

Figure 4.5: *Performance comparison of the performance of recognition model using GLOH-color combination (GC) on the Caltech 4 dataset.* Two different counter-classes are used separately: the Graz01 background and (GC-G) the Caltech-leaves datasets (GC-L).



(a)

(b)

Figure 4.6: *Example of incorrectly classified images on the Caltech dataset using the GLOH-color combination (GC-G).* (a) Example of false negative images classified as background from the classes motors (first row), airplanes (second row) and faces (third row). (b) Example of false positive images classified as motors (first row), airplanes (second row) and faces (third row).

Table 4.4: Classification rates at ROC-eqq.-err. rates of our recognition model using the Caltech 4 dataset compared to state-of-the-art approaches.

Class	GC-G	GC-L	[114]	[43]	[143]	[157]	[136]	[15]	[19]	[45]	[162]	[29]
Motors	96	96	92.2	92.5	93.2	88	84.6	95	93.3	93.3	95	-
Cars	100	94	91.1	90.3	90.2	86.5	78.6	96.9	97.7	99.3	-	97.8
Air-planes	84	82	88.9	90.2	83.8	-	96.6	95.6	89.7	95.3	94.4	-
Faces	94	98	93.5	96.4	83.1	93.5	94.7	89.5	92.1	83	99.7	92.4
Avg. over used classes	93.50	92.50	91.43	92.35	87.56	89.33	88.63	94.25	93.20	92.73	96.37	95.10

4.3.2 Experiments using Graz02 Dataset

Further evaluation experiments of our recognition approach are carried out using Graz02 dataset. Graz02 dataset is more difficult than Caltech 4. Objects are shown on complex cluttered background, at different scales and with different object positions. The images include high amount of occlusion up to 50% [114]. It is also balanced with respect to background so it is not possible to detect an object on its context *e.g.* cars by traffic sign [104].
For the experiments using this dataset, we use 150 positive and 150 negative images for training (total of 300 images) and 75 positive and 75 negative previously unseen images for testing (total of 150 images) as in [114]. The features of each image are clustered to 100 cluster centers using the k-means clustering algorithm and the learning procedure is run using $T = 150$ to give more possibility to generalize over the complex data [114].
The counter class provided by the Graz02 dataset is of colored images. Therefore, there is no need here to find another counter-class than the one provided by the dataset.
Table 4.5 displays the results of the recognition using each descriptor separately as well as combined while figure 4.7 displays the ROC curves of the three used classes using the GLOH-color combination (GC). The results in table 4.5 reveal the difficulty of the Graz02 dataset (as previously mentioned) when compared to the results achieved on the Caltech 4 dataset (tables 4.2 and 4.3). The GLOH-color combination (GC) yields performance gain for the classes ”cars” and ”persons”, while does not add any gain for recognition performance of the class ”bikes” when compared to the results using each GLOH descriptor separately. One reason for this could be that the use of the color features with the GLOH does not add any significant information for improving the recognition performance over that is achieved by GLOH alone. However, the use of color could have added some redundancy which has affected in turn the performance of their combination.
Table 4.6 provides a comparison of the recognition performance of our approach to the performance of other approaches that used the same dataset for evaluation. The comparison reveals that our model achieves the best performance on the class ”cars” while achieves the second best on the class ”persons”. The last column of Table 4.6 displays the overall average performance of each approach (computed as proposed in section 4.3.2). We can then notice the robust performance of our model.
It should be noted that the amount of supervision used by the approaches in table 4.6 is not the same. Approaches in [114, 105, 104] as well as ours used only class labels for training (weakly supervision) while [108] used class labels and localization ground-

Table 4.5: Classification rates at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using GLOH, color and GLOH-color combination (GC) respectively.

Class	GLOH	Opponent angle	Combination
Bikes	78.67	73.33	74.67
Cars	81.33	74.67	81.33
Persons	77.33	78.67	81.33

truth for training. Moreover, the number of training examples used by each approach varies. Our approach and the one in [114] use the same number of training images per class (150 images) while the approaches mentioned in [105, 104] use 300 training images per class which is double the number we use. The approach in [108] uses only 50 training images per class.

Figure 4.8 displays example of incorrectly classified images which are classified either as false negatives (figure 4.8 (a)) or as false positives (figure 4.8 (b)). As we can notice from the figure that the failure in classification in the false negatives case is either due to high degree of occlusion ((figure 4.8 (a)) second row, left image) or small size of objects within images. For the false positives, some incorrect classification could be due some plausible reasons such as figure 4.8 (b), first row, left image where the image is classified as containing an object of class bikes or figure 4.8 (b) last row, right image where a person photo in the wall made the image to be classified as containing an object of class persons.

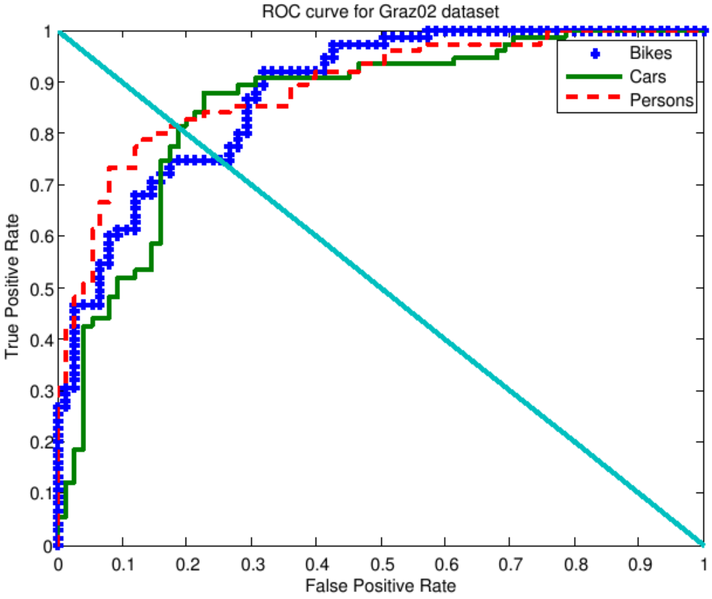
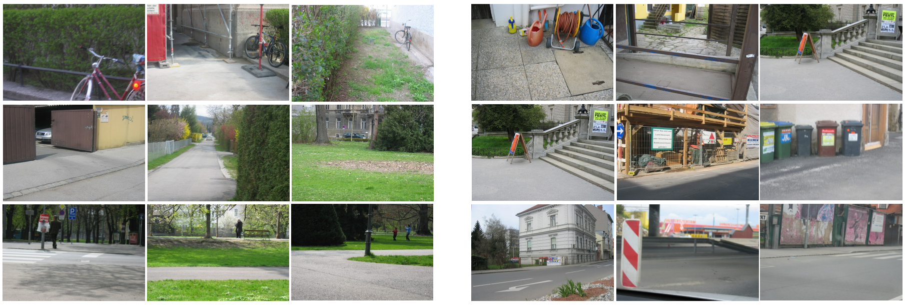


Figure 4.7: ROC curves of the performance of the recognition model using the GLOH and color combination (GC) on the Graz02 dataset.

Table 4.6: Classification rates at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using the GC combination compared to the state-of-the-art approaches.

Class	GC	Opelt[114](Comb.)	[108]	[105]	[104]
Bikes	74.67	77.80	80.50	84.40	79.9
Cars	81.33	70.5	70.10	79.90	71.7
Persons	81.33	81.2	81.7	-	-
Avg. over classes	79.11	76.5	77.43	82.15	75.8



(a) (b)

Figure 4.8: *Example of incorrectly classified images on the Caltech dataset using the GLOH-color combination (GC-G).* (a) Example of false negative images classified as background from the classes bikes (first row), cars (second row) and persons (third row). (b) Example of false positive images classified as bikes (first row), cars (second row) and persons (third row).

4.4 GLOH vs. SIFT: Performance Evaluation

As mentioned in section 4.2.3, GLOH descriptor is the extension of the publicly used SIFT descriptor and has shown better performance than SIFT in many evaluations in the context of image matching and retrieval (*e.g.* [101]). We aim by the following experiments to provide a performance comparison of the two descriptors on a different

context, namely the context of generic object recognition.
To establish a comparison between the performance of both GLOH and SIFT descriptors, we will repeat all the previous experiments replacing the GLOH descriptors with the SIFT.

4.4.1 Using Caltech 4 Dataset

The recognition results using the SIFT, color and SIFT-color combination (SC) respectively on the Caltech 4 dataset are displayed in tables 4.7 and 4.8 for the two different counter-classes mentioned in section 4.3.1. The recognition performance of the model using SIFT-color ((SC-G) and (SC-L)) is robust. The use of SIFT improves the recognition performance of almost all the classes when compared to the recognition results using GLOH (tables 4.2 and 4.3). Figure 4.9 displays a comparison of the ROC curves of the recognition results using GLOH-color (GC-G) and SIFT-color (SC-G) combinations. Although GLOH showed better performance than SIFT in the context of image matching and retrieval (as shown in [101]), our results reveal that SIFT exceeds GLOH in performance in the context of GOR.
Table 4.9 compares the performance of the our recognition results using the SIFT-color combination (SC-G) to the state-of-the-art approaches. It is clear from the table that the use of SIFT improved the recognition performance of our model (when compared to table 4.4. Our model (SC-G) achieves superior performance on the classes motors, cars and faces.

4.4.2 Using Graz02 Dataset

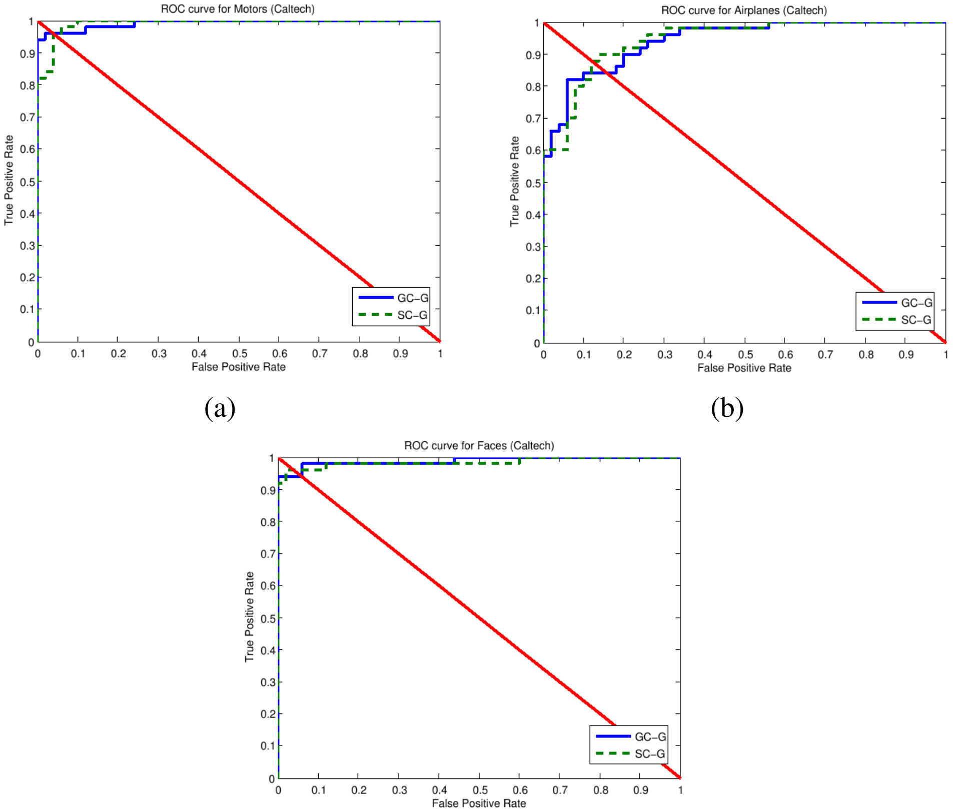
The same conclusion can be drawn when Graz02 dataset is used. Table 4.10 displays the results of the recognition model when the SIFT descriptors are used. Their combination with color (SC) increased the recognition performance on both the motors and persons datasets while the performance on the cars dataset is decreased (with respect to the performance of the GC combination). When compared to the model using GLOH (GC) (see Table 4.11 and Figure 4.10), the results using (SC) reveal a better performance.

Table 4.7: Classification rates at ROC-eqq.-err. rates of our recognition model using SIFT-color combination (SC-G) on the Caltech 4 dataset with the Graz01 background class as a counter-class.

Class	SIFT	Opponent angle	Combination (SC-G)
Motors	90%	94%	**96**%
Cars	86%	100%	**100**%
Airplanes	78%	80%	**88**%
Faces	92%	86%	**96**%

Table 4.8: Classification rates at ROC-eqq.-err. rates of our recognition model using SIFT-color combination (SC-L) on the Caltech 4 dataset with the leaves class as a counter class.

Class	SIFT	Opponent angle	Combination (SC-L)
Motors	86%	82%	**93**%
Cars	90%	86%	**94**%
Airplanes	78%	80%	**84**%
Faces	96%	94%	**100**%



(a) (b)

(c)

Figure 4.9: Comparison between the performance of the recognition model using GC combination and the SC combination on the Caltech4 dataset using the Graz01 background class as a counter-class.

Table 4.9: Comparison between the performance of the recognition model using (GC-G) combination and the (SC-G) combination on the Caltech 4 dataset compared to state-of-the-art approaches.

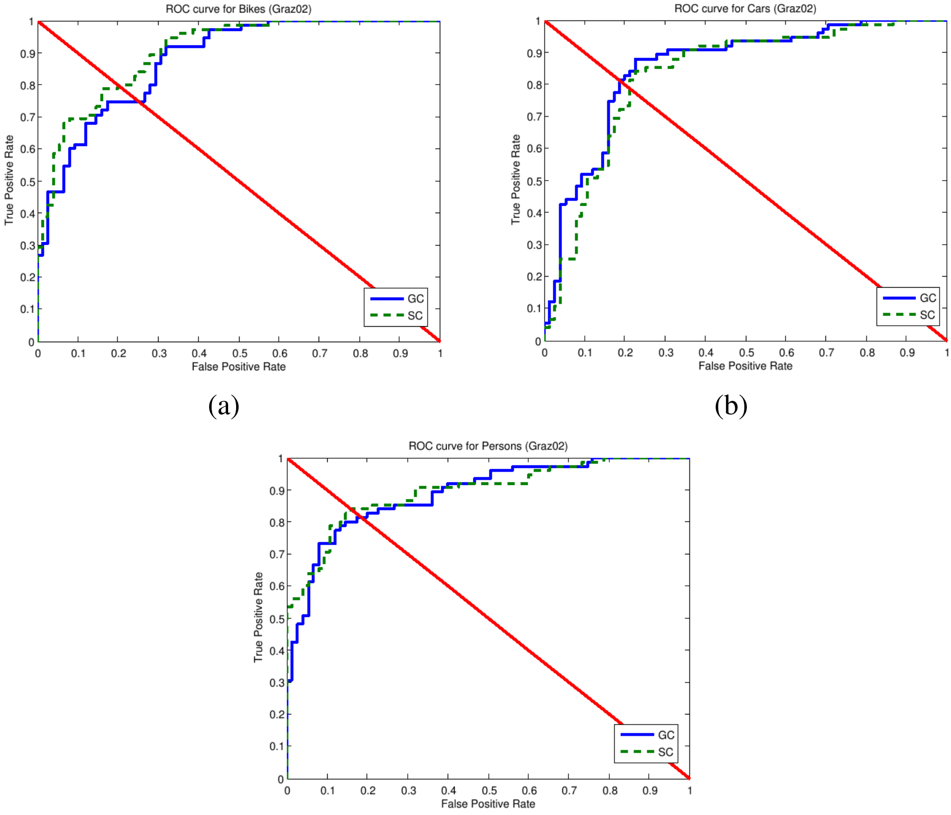
Class	GC-G	SC-G	[114]	[43]	[143]	[157]	[136]	[15]	[19]	[45]	[162]	[29]
Motors	96	96	92.2	92.5	93.2	88	84.6	95	93.3	93.3	95	-
Cars	100	100	91.1	90.3	90.2	86.5	78.6	96.9	97.7	99.3	-	97.8
Airplanes	84	88	88.9	90.2	83.8	-	96.6	95.6	89.7	95.3	94.4	-
Faces	94	96	93.5	96.4	83.1	93.5	94.7	89.5	92.1	83	99.7	92.4
Avg. over used classes	93.50	95.00	91.43	92.35	87.56	89.33	88.63	94.25	93.20	92.73	96.37	95.10

Table 4.10: Classification rates at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using SIFT, color and SIFT-color combination (SC) respectively.

Class	SIFT	Opponent angle	Combination
Bikes	78.67	73.33	80.00
Cars	74.67	74.67	77.33
Persons	78.67	78.67	82.67

Table 4.11: Classification rats at ROC-eqq.-err. rates of our recognition model on the Graz02 dataset using the SC combination compared to the state-of-the-art approaches.

Class	GC	SC	Opelt[114](Comb.)	[108]	[105]	[104]
Bikes	74.67	80.00	77.80	80.50	84.40	79.90
Cars	81.33	77.33	70.5	70.10	79.90	71.70
Persons	81.33	82.67	81.2	81.7	-	-
Avg. over classes	79.11	80.00	76.5	77.43	82.15	75.8



(a) (b)

(c)

Figure 4.10: Comparison between the performance of the recognition model using GC combination and the SC combination on the Graz02 dataset.

4.5 Evaluation of Boosting Algorithms on GOR

AdaBoost algorithm is used in the learning procedure of our model in all previous experiments. The advantages of the AdaBoost algorithm as mentioned in section 2.2.1 include that it is the most popular boosting algorithm, generates combined hypotheses with large margins and works well on data with low noise. However, AdaBoost is reported by different studies low generalization performance with the presence of data with high degree of noise.
SoftBoost algorithm is a newly presented boosting algorithm [154], which implements the idea of soft margins exists in the SVM in practical boosting algorithm. Details of the algorithm are given in section 2.2.1
The main objective of this section is to investigate the recognition performance of the SoftBoost algorithm on GOR as, to the best of our knowledge up to now, no evaluation of this new algorithm exist on a real world object recognition problem. An additional objective is to evaluate the performance of SoftBoost algorithm by comparing it to the performance of AdaBoost algorithm using label noisy training data [3].
Discrete AdaBoost [50] (see algorithm 1) and AdaBoost with confidence-rated prediction [133] (Real Adaboost) (see algorithm 2) are used in the evaluations performed in here.
For the Discrete AdaBoost, the weak learner presented in section 4.2.4 is used. For the AdaBoost with confidence-rated prediction, a weak learner which delivers a weak hypothesis $h : \chi \rightarrow \mathbb{R}$ (in our case $h : \chi \rightarrow [-1, +1]$) is required.

4.5.1 Used Weak Learner

The used weak learner is the same as the one described in section 4.2.4 . However, we applied some modification to make it similar to the idea of the weak learner used by the authors of SoftBoost algorithm [154]. This modification is that, in algorithm 5, step 4, the description vectors $(t_{i,f}, v_{i,f})$ which delivers the minimum classification errors on the training data are selected. The classification outpout of selected weak hypothesis from each type is computed as follows:

---

[3] The work presented in this section is adapted from our paper [60]

$$f_k^x = \begin{cases} \dfrac{\left(\theta_k^x - d\left(v_k^x, I\right)\right)}{\theta_k^x} & \text{if } d\left(v_k^x, I\right) < \theta_k^x \\ \\ \dfrac{\left(\theta_k^x - d\left(v_k^x, I\right)\right)}{M - \theta_k^x} & \text{Otherwise.} \end{cases} \tag{4.7}$$

where M is a distance maximum bound used to compute the scores of the hypotheses output and to make $f_k^x \in [-1, 1]$. M is selected as the maximum distance computed from step 2 of algorithm 5. This results in a classification output for each weak hypothesis $f_g, f_c \in [-1, 1]$. Then the combination output of the two chosen different weak hypotheses is computed as follows:

$$h_k = \begin{cases} \max(f_g, f_c) & \text{if } f_g \text{ or } f_c > 0 \\ \\ \min(f_g, f_c) & \text{Otherwise.} \end{cases} \tag{4.8}$$

4.5.2 Experiments and Results

To investigate the recognition performance of the SoftBoost algorithm, a set of experiments is performed using it as the base learning algorithm in the recognition model. Moreover, some experiments using Discrete AdaBoost and Real AdaBoost algorithms are performed in order to establish a performance comparison and evaluation of the three algorithms.

Experimental settings

The experimental settings are divided as follows:

- The used dataset: Graz02 dataset [5] is used in all evaluation experiments.
- Training and Test images: For training, 300 examples (images) are used: 150 images of object class and 150 images of counter-class (negative). For testing, 150 previously unseen examples are used: 75 images of object class and 75 images of the counter-class.
- SoftBoost algorithm: The value of guarantee g (see section 2.2.1) is set equal to γ^* in all experiments, where γ^* is the value of linear programming problem $P2$ presented in [154]. The value of accuracy parameter δ used in all experiments is 0.0001. This value is used based on performing experiments on the bikes class, with values of

$\delta \in \{0.1, 0.01, 0.001, 0.0001, 0.00001\}$. The best generalization performance is achieved at $\delta = 0.0001$.

- **AdaBoost algorithms:** The AdaBoost algorithms are run for a number of iterations T= 150.

Based on the results provided in the previous section, all experiments are performed using our recognition model with the SIFT-color combination (SC).

Experiments Using Noise Free Data

Using SoftBoost Algorithm: an important parameter of the SoftBoost algorithm is the capping parameter ν. It represents the number of training examples that are allowed to have wrong predictions in order to obtain high generalization performance. Therefore, the optimal value of ν should be selected. This is accomplished by using a 5-fold cross-validation for each object class. This results in estimating three optimal values of ν, one for each object class. Training is performed afterwards using the selected values of ν and the generalization rates are given in table 4.12.

Using AdaBoost Algorithms: Training the three object classes using the two AdaBoost algorithms is performed using T iterations specified in section 4.5.2 and the resultant generalization rates are shown in table 4.12. When comparing SoftBoost, Discrete AdaBoost (D. AdaBoost) and Real AdaBoost (R. AdaBoost) algorithms, it is clear that Real AdaBoost outperforms the other algorithms in two of the three object classes (cars and persons) while SoftBoost and D. AdaBoost algorithms achieve higher generalization rates on the bikes class. Moreover, the average performance over the classes of the recognition model using the three different boosting algorithms is almost the same (see table 4.12, last column). Few performance gain is achieved by using the SoftBoost algorithm.

Experiments Using Noisy Data

Actually, *noise* could be observed at various levels of abstraction in learning and recognition process, including high intra-class variability, partial occlusion, background clutter, varying illumination and added Gaussian noise to the test images. In fact, Graz02 database is already quite noisy in this respect. Additionally, more difficulties could be added to the training data by the presence of outliers or miss-labeled (label noisy) patterns. Label noise means that a pattern is clearly a member of one class and its label corresponds to the alternate class. The label noisy patters or examples cause the

Table 4.12: Generalization performance of Discrete AdaBoost, Real AdaBoost and SoftBoost algorithms represented by classification rates ROC-eqq.-err. rates.

Class	D. AdaBoost	R. AdaBoost	SoftBoost	optimal ν (SoftBoost)
Bikes	80.00	76.00	80.00	40%
Cars	77.33	80.00	78.67	70%
Persons	82.67	84.00	82.67	30%
Avg. over classes	80.00	80.00	80.45	

boosting algorithm to concentrate on them during training, which in turn, deteriorates the final hypothesis and thus the generalization performance of the algorithm.

In this set of experiments, label noise is added by assigning wrong labels to a percentage n of the training examples, where $n = 10, 30$ and 50% respectively. This means that three different degrees of label noise are given. The test examples are left unchanged. For training using SoftBoost algorithm, a 5-fold cross-validation is used to select the optimal value of ν at each level of noise for each class. This results to the estimations of nine values of ν.

Table 4.13 presents the generalization rates of AdaBoost and SoftBoost algorithms on this set of experiments. Generally, the performance of SoftBoost exceeds the performance of AdaBoost algorithms when the training data contains high degree of label noise (i.e. 50%). At small degrees of label noise (i.e.10% and 30%), AdaBoost algorithms, especially R. AdaBoost, achieve better performance.

However, the use of SoftBoost did not significantly improve the recognition performance. The performance gain we could get from using the algorithm could be in the cases when its known in advance that the data has high degree of noise (e.g. more than 50%). In this case, the use of SoftBoost could be profitable. Otherwise, its time and effort consuming task to use the algorithm as a proper value for the capping parameter ν should be then investigated. For achieving this, many different trails (cross-validations) should be performed.

Table 4.13: Generalization rates (classification rates at ROC-eqq.-err. rates) of Discrete AdaBoost, Real AdaBoost and SoftBoost algorithms using training data with different degrees of label noise.

Class	noise degree	D. AdaBoost	R. AdaBoost	SoftBoost	optimal ν (SoftBoost)
Bikes	10%	78.67	78.67	74.67	70%
	30%	68.67	74.67	74.67	70%
	50%	57.33	60.00	58.67	30%
Cars	10%	77.33	77.33	70.67	30%
	30%	69.33	76.00	73.33	30%
	50%	60.00	60.00	68.00	50%
Persons	10%	78.76	82.67	82.67	40%
	30%	68.00	80.00	77.33	30%
	50%	48.00	52.00	66.67	50%

4.6 Conclusions

In this chapter, we have presented our model for generic 2D object recognition. The model exploits a combination of appearance-based information (descriptors) for binary learning and classification of generic object classes. Two different types of descriptors are used, texture and color descriptors, as they are important information for describing the appearance of objects. Boosting, namely the AdaBoost algorithm, is used for learning the different descriptions of objects in a weakly supervised manner. Images that show the objects at varying scale and viewpoint in highly cluttered background are used in the learning step as well as for evaluating the model. Using such difficult data, we have experimentally illustrated the robustness, with respect to classification, of our model which does not make use of any geometrical or spatial information for learning and recognition. Additionally, comparisons with other approaches, including state-of-the-art approaches, have been presented and they have revealed the strength of our recognition model.

In our experiments, we have evaluated small but important aspect for recognition. It is the choice of the counter-class used for binary classification. Some datasets (e.g.

Caltech) provide a counter-class that is considered to be an easy one. The contexts of its images are totally different from those of the object classes which makes the recognition in this case an easy task. Moreover, it does not reveal the drawbacks of the recognition model as all the images could be classified correctly because the images of counter-class has completely different context from the images of the object class. In our experiments, we have (fortunately) use different counter-classes in the experiments using the Caltech dataset. This helped us to investigate the importance of the counter-class and reveal its importance for building and correctly evaluating a recognition model.

Moreover, in our experiments we compared the performance of two famous texture descriptors, namely SIFT and GLOH descriptors, in the context of GOR. SIFT revealed shown better performance in this context than GLOH, although GLOH is reported to yield better performance, but in the context of image matching and retrieval. Therefore, careful selection of the description methods and tools is required, as the performance of descriptors vary from context to another. For the GOR, it is more complicated as the performance of one descriptor varies form an object category to another.

Finally, we have presented a performance evaluation of a new Boosting algorithm, the SoftBoost algorithm. Theoretically, the algorithm has some good advantages which seem to overcome some of the drawbacks exist in popular and famous boosting algorithms such as the AdaBoost. Practically, however, the algorithm did not reveal the expected performance. It is designed to deal with data which contains high degree of noise. However, this information should be known before hand to be sure that the Soft Boost algorithm is the only suitable algorithm for the problem. Otherwise, using the algorithm would be a time and effort consuming process.

Chapter 5

A 2D/3D Object Category Dataset

This chapter introduces a new generic object recognition (GOR) dataset which has the advantage over all existing datasets of providing range data in addition to 2D data about different object classes. A brief description of the 3D range image acquisition methods, including the method used to construct the dataset, is first given. A description of the dataset and its different member classes is then presented. Afterwards, the acquisition procedures of the dataset are explained. Moreover, an experimental evaluation of the dataset as well as comparison to other benchmark (Graz02) using our recognition model is established.

5.1 Motivation

Several datasets have been emerged as standards for the GOR community such as UIUC [3, 13] Caltech-4 [2, 43], Caltech 101 [9, 41] and Graz [4, 5, 113] and other datasets presented in section 2.4. However, all these datasets provide only 2D data (images) of their object classes and none of them provide range images bout their members. Up to our knowledge, there is no available GOR dataset which provides range data of different object classes and, hence is suitable for GOR . The lack of such dataset is one reason that most of the current state-of-the-art GOR approaches depend on only information provided by 2D images in recognition (e.g. [43, 114, 13]) and ignored or neglected the use range data.

Range images are very useful for GOR because they have the advantage of providing direct information about the shape of objects. This shape information is suitable for recognition of generic object class from their shapes as well as for generic 3D object recognition. Moreover, range information depends only on geometry and is independent of illumination and reflectivity and intensity-image problems with shadow and surface markings do not occur. Therefore, the process of recognizing objects by their shape should be less difficult in range images than intensity images [25]. The current range images datasets such as Stuttgart Range Image Database [8, 63] (see section 2.4.3) are suitable only for specific 3D object recognition and are used normally for this task.

With the upcoming technique of Time-of-Flight cameras (TOF), for example the PMD-cameras , range images can be acquired in real-time and thus, recorded range data can be used for building a GOR dataset.

This chapter presents a new object category dataset which provides two different image types of different object classes: 2D and range images. Actually two versions of the dataset has been constructed. The first version (JenaRange01) provides 2D (intensity images) and 3D range images. It contains small number of classes and small amount of background clutter and occlusion. The second version (JenaRange02) provides 2D (color images) in addition to the range images. It contains more object classes, provides more real images full of background clutter, occlusion, size and view point variations and large amount of intra and inter-class variabilities. Therefore, our dataset facilitates the recognition of generic 2D as well as 3D objects from range images or from intensity and color images as well. Furthermore, a combination of all three different image types could be exploited for more robust GOR. The dataset is the main contribution in this chapter and is available for public use [1] as we believe that the dissemination and use of this dataset will allow realistic comparative studies as well as a source to test data for development for new techniques of GOR from range images.

In section 5.2, we briefly discuss the different range imaging techniques with presenting a brief overview of the technique used for constructing our dataset (TOF). Section 5.3 presents the JenaRange01 dataset. It provides a description to the dataset, its member classes and its properties as well as mentioning its construction procedures. In section 5.4, the JenaRange02 is presented. Experimental evaluations of our dataset are accomplished in section 5.6 to investigate its suitability for the GOR problem.

5.2 3D Image Acquisition Methods

There exist three main methods for 3D images acquisition [28]:

---

[1] JenaRange01 is available at: http://www.inf-cv.uni-jena.de/index.php?id=dataset.

1) *Triangulation*: the acquisition of 3D images without scanning components is done by stereoscopy (stereo vision), meaning the combination of two 2D images from different viewpoints. The geometrical relations between the object, the sensor and a known basis line are used for the calculation of the distance. The processing of the distance maps (images) is based on correlation calculations that are very time-consuming [28]

2) *Interferometry*: constructive and superpositions superpositions of at least two light beams are exploited for measuring very small differences of distances in the sub-micrometer range. Accordingly, the accuracies are very high and depend mainly on the coherence length of the light source. Real-time 3D image acquisition is possible using CMOS photo sensors. Thereby, the imaging process is reduced from three scanning directions to just one direction. However, interferometry is not suitable for ranges starting from centimeters up to several meters as the method is based on the evaluation of the very short optical wavelength [28] .

3) *Time-of-Flight*: is based on the measurement of the time the light needs to travel from the lighting source to the object and returns back to the sensor (see figure 5.1). This method is suitable for ranges starting from few centimeters to several hundreds of meters.

Stereo vision is the most common and well-known principle for 3D imaging principle, which is known and used for long time for 3D image acquisition in computer vision community [66]. The advantages of the stereo vision over other 3D imaging techniques (devices) such as laser scanners or radar sensors is that it achieves high resolution and simultaneous acquisitions of the entire range images without energy emission or moving parts [66]. However, stereo vision suffers from some disadvantages such as the limited field of view and the correspondence problems.
In the past years, modality of Time-of-Flight (TOF) imaging became more and more attractive to the research community due its advantages which overcome some of the limitations and disadvantages exist in stereo vision and other 3D image acquisition techniques. Example of the TOF advantages is providing direct depth data acquisition instead of requiring long time of computation to produce the same 3D images using other techniques (*e.g.* stereo vision). Authors of [66] provide a deep comparison between the stereo vision and the TOF (PMD) techniques. Figure 5.2 is been borrowed from [66] and summarizes the disadvantages of each technique. Further details could be found in [66]. Following, we give a brief description of the Time-of-Flight (TOF) PMD camera used for acquiring the range images of our GOR dataset.

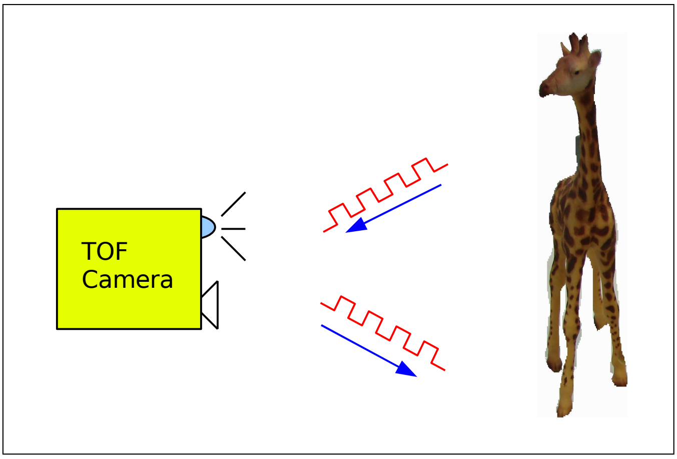


Figure 5.1: Illustration of the principle of the Time-of-Flight rangers.

5.2.1 Time-of-Flight PMD Camera

A new and promising approach developed in recent years estimates the distance by Time-of-Flight (TOF) measurements for modulated, incoherent light based on the new Photo Mixing Device (PMD) technology [79]. This new camera technology realizes 3D imaging without complex electronics and without scanning with a solid state images similar to a CMOS device [126]. As previously mentioned, other techniques for providing 3D depth maps of the scene need high computer power to find correlation in the gray value map like stereo cameras. Some techniques require the existence of mechanical components like scanning systems. Both, stereo cameras and scanning systems, are cost-intensive, have a low realtime capability and have no homogeneous depth map (in the case of stereo cameras) [126].

The principle of the range measurement in a TOF PMD camera is based on the TOF principle mentioned previously. However, rather than using a single laser beam, the whole scene is illuminated with modulated light. The camera sends out modulated light and measures its reflections on the scene. Within each pixel, this reflected signal is mixed with the sent out signal to obtain the phase shift, as shown in figure 5.3. This phase shift can then be easily translated to the distance between the pixels and the different points in the scene [109].

The TOF (PMD) technique has many advantages:

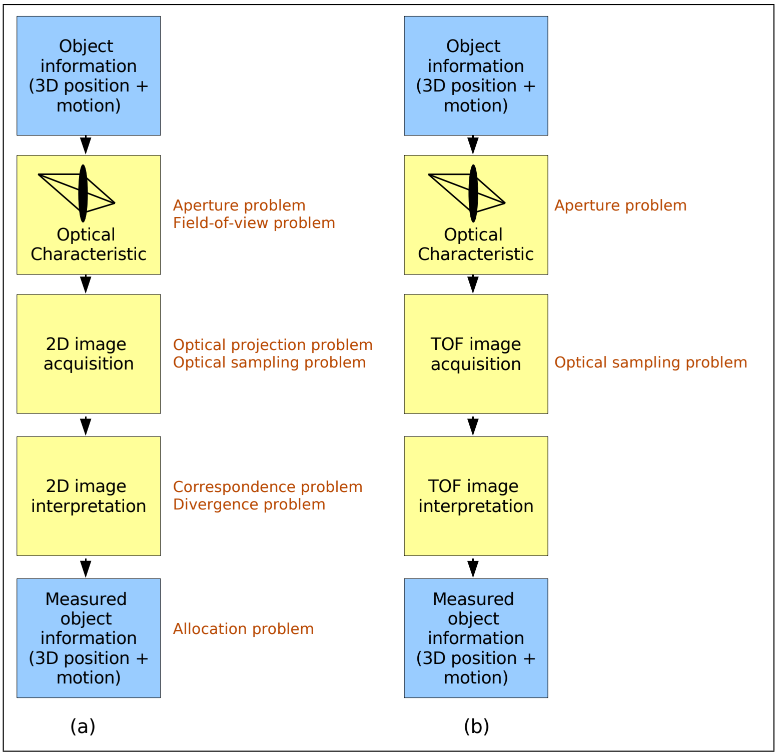


Figure 5.2: Functional block diagram of the image processing chain of (a) Stereo vision systems and (b) TOF vision systems [66].

1. Capable of delivering complete images at once where no scanning is needed.
2. Acquiring realtime images is possible.
3. Does not have moving components.

However, it suffers from some drawbacks such as the degradation of accuracy proportionate to the background light and the limited distance range due to the periodicity of the modulation signal [109].
A PMD 19k 3D camera is used in acquiring the different views of our 3D object category dataset. The PMD 19k camera provides images of 160 × 120 pixels. Each pixel

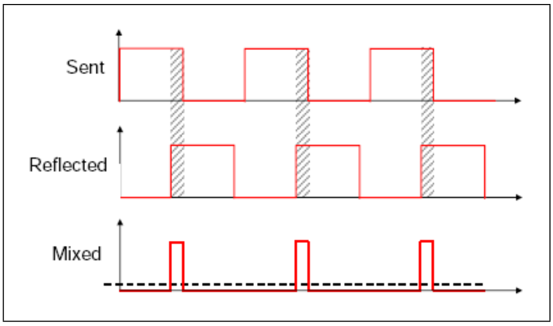


Figure 5.3: Phase measurements by averaging mixed reflected and sent-out waveforms [109].

delivers distance information, distance resolution and grayscale information of the acquired scene. However, the delivered intensity image is of low resolution, which affects the direct application of some image processing algorithms on them.

5.3 JenaRange01 2D/3D Dataset

An object category dataset of 936 2D/3D images (2D grayscale as well as range data) of 26 objects (36 images per object) is built using a 3D Time-of-Flight PMD camera. The objects are instances of three main visual categories (classes): cars, motors and animals (see figure 5.4). A fourth class is constructed to be used as a background or a negative class during training and testing. This background class consists of objects that are visually different from the object instances of the three main classes.
Due to the difficulty to record different outdoor views of real objects using the PMD camera [2], human made objects (artificial objects) are used to build the dataset. The instances of each object class are chosen with different size and appearance to achieve large intra-class variability as much as possible as shown in figure 5.4.

---

[2] Settings required to use a PMD camera make it difficult to acquire outdoors views of real objects.

5.3.1 Dataset Acquisition Process

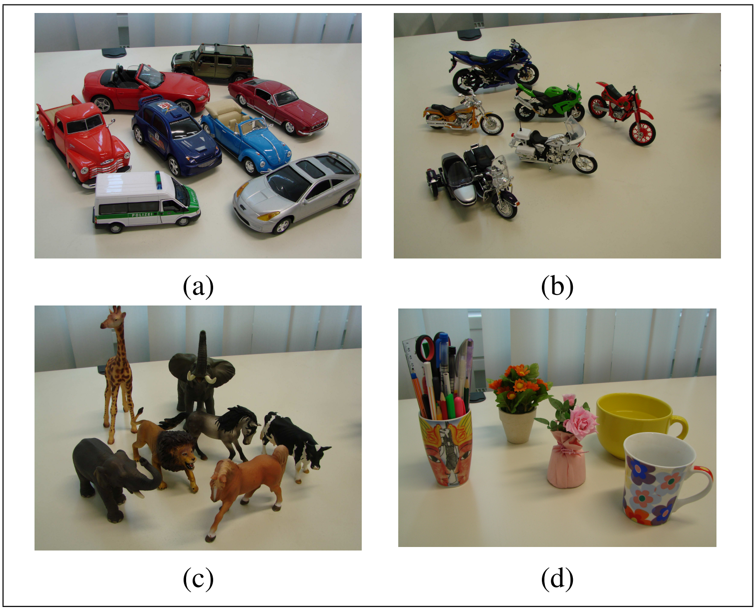
A sketch diagram of the acquisition procedure is displayed in figure 5.5. A 3D PMD camera was fixed to a rigid stand about 1.1 meters from its base. A motorized turntable was placed about 2 meters from the base of the stand. It is noticed by experiments that, by placing the turntable closer than 2 meters from the camera, the resultant images contain inaccurate distance measurements. For this reason, the size of the objects within the images is relatively small. The camera was set in a way that the objects appear in the center of the image when placed at the center of the turntable. White background was provided by placing the turntable in front of a white wall. The normal lighting condition of the room was used.

Each object was placed in a stable configuration at approximately the center of the turn table. The turntable was then rotated through 360 degrees about the vertical axis and 36 2D/3D images were acquired per object; one at every 10 degrees of rotation. Figure 5.6 shows different example 2D images of the three classes.

Figure 5.7 displays the distribution of the mean of range data of each image in the dataset for the three classes. It reveals that the dataset is not a trivial one despite the relative small size of the object within the images.

The range data acquired by the TOF PMD camera are stored in binary files with the extension ”pfm”. Each pixel in the range image is represented by three values: the range distance, the precision of the range distance and the intensity value of the pixel. The images acquired by the PMD camera are of resolution 160 × 120 pixels. An example of image filename of the category car is ”car100090020.pfm” (range data binary file);

- ”car” indicates the category.
- ”1” indicates the object instance number.
- The last two digits ”20” indicates the angle of rotation about the vertical axes.



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

Figure 5.4: *The three visual classes of the new 3D object category database JenaRange01*. (a) Cars, (b) motors, (c) animals and (d) background (counter-class).

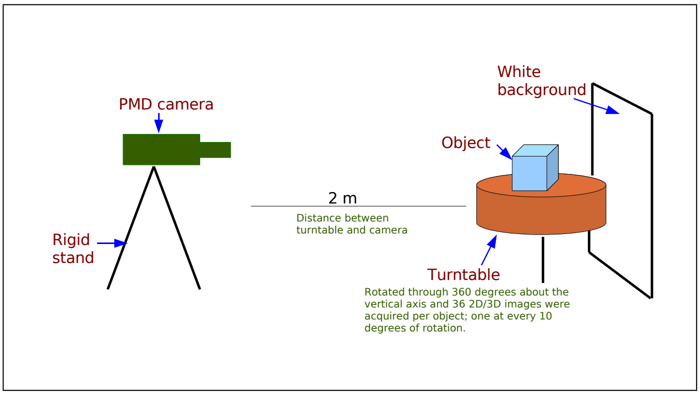
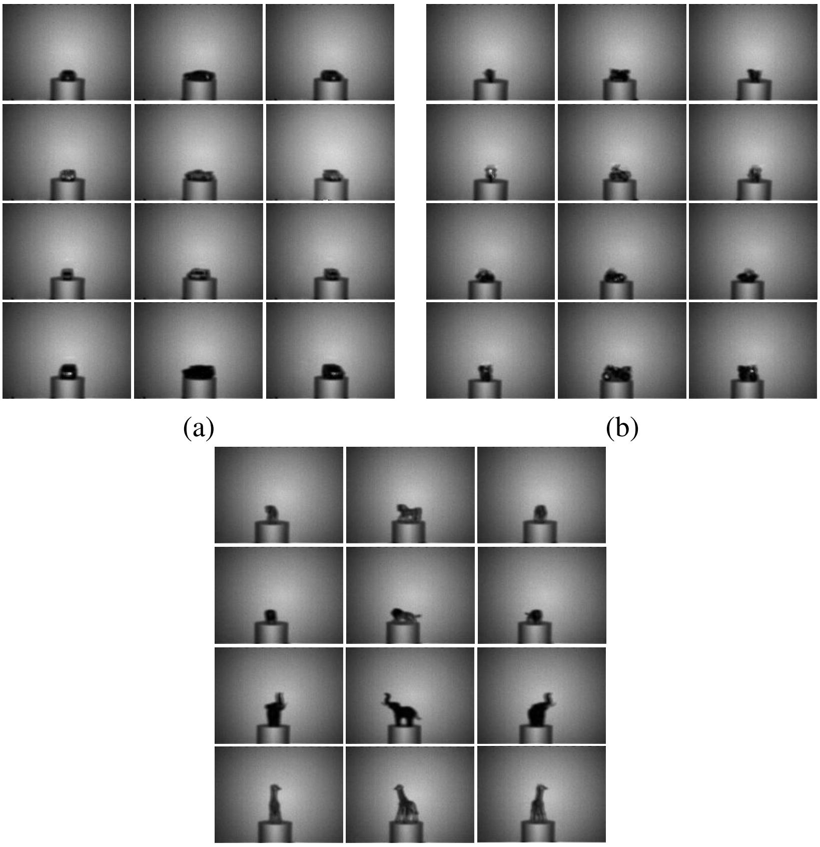

White background
PMD camera
Object
2 m
Distance between turntable and camera
Rigid stand
Turntable
Rotated through 360 degrees about the vertical axis and 36 2D/3D images were acquired per object; one at every 10 degrees of rotation.

Figure 5.5: The acquisition procedure of the dataset images.



(a) (b)

(c)

Figure 5.6: *Example 2D images of the dataset for the three visual classes.*(a) Cars, (b) motors and (c) animals.

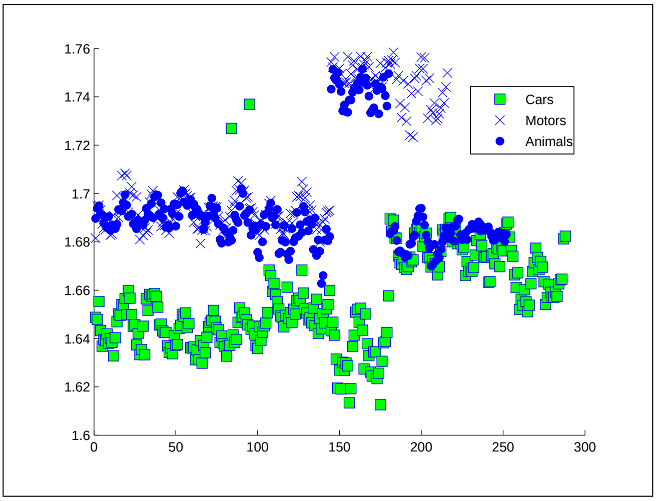


Figure 5.7: Distribution of range means of the dataset images for the three object classes.

5.4 JenaRange02 2D/3D Dataset

A new version of the dataset is constructed and described here. This new dataset is of 4220 2D/3D images (2D colored and range images) of 35 objects and was constructed using a 3D Time-of-Flight (TOF) PMD camera [79] and a CCD camera. The objects are instances of five visual classes (categories): cars, toys, cups, fruits and animals (see Figure 5.8). For each object category, seven individual instances were used. Due to the difficulty to record different outdoor views of natural objects using the PMD camera, indoor views in an office environment were captured. Artificial objects were used in replacement of real instances of some visual classes (namely cars and animals) in building the dataset. The instances of each object class were chosen again with different size and appearance to achieve large intra-class variabilities as much as possible (see Figure 5.8). Many images of the dataset contain multiple instances of the same class or from different classes. Moreover, the images contain large viewpoint and orientation variations, partial occlusion (e.g. by other objects) and truncation (e.g. by the image boundaries) as well as background clutter (see Figures 5.9 and 5.10) .

5.4.1 Dataset Acquisition Process

The images of the dataset were acquired using two different cameras, a PMD camera and a CCD color camera which had nearly the same field of view (see Figure 5.9). This results in two different types of images (color and range) for the same scene as shown in figure 5.11. The images of each individual object instance were acquired under eight different viewing angles and four different heights as shown in figure 5.9. This is accomplished as follows: at each height, each object instance was placed on a turn table which was rotated through 360 degrees about the vertical axis and eight colored and range images were acquired; one at every 45 degree of rotation. The total number of images acquired using each camera is 32 images for each object instance (4 heights $\times$ 8 angles). The PMD camera delivers also an intensity image corresponding to each range image.

The range data are stored in binary files with the extension ”pfm”. The value of each pixel in the range image is represented by three values: the range distance, the precision of the range distance and the intensity value of the pixel. The images acquired by the PMD camera are of resolution 160×120 pixels. For the colored images, they are stored in ”jpg” formate and are of the size 640×480 pixels.

An example of image filename of the category car is ”c1125.jpg” (colored image) or ”c1125.pfm” (range data binary file);

- ”c” indicates the category.
- ”1” indicates the object instance number.
- ”1” indicates the viewing distance (scale).
- ”2” indicates the viewing height.
- ”5” indicates the viewing angle.

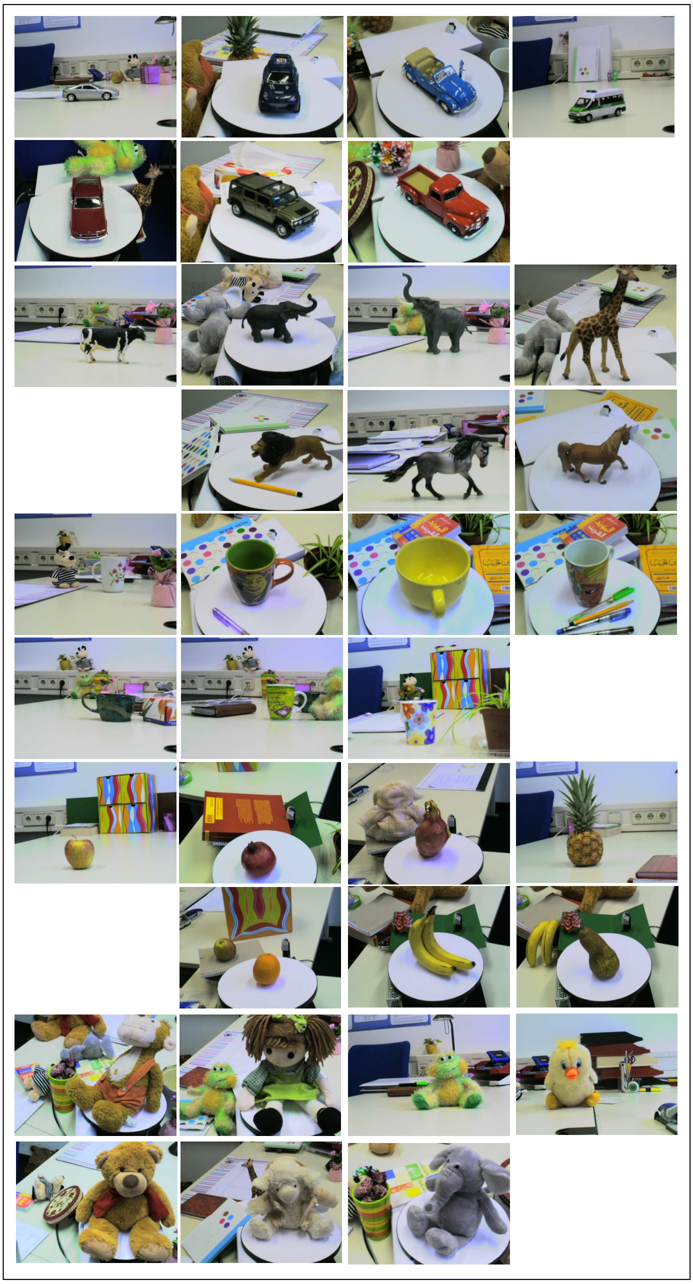
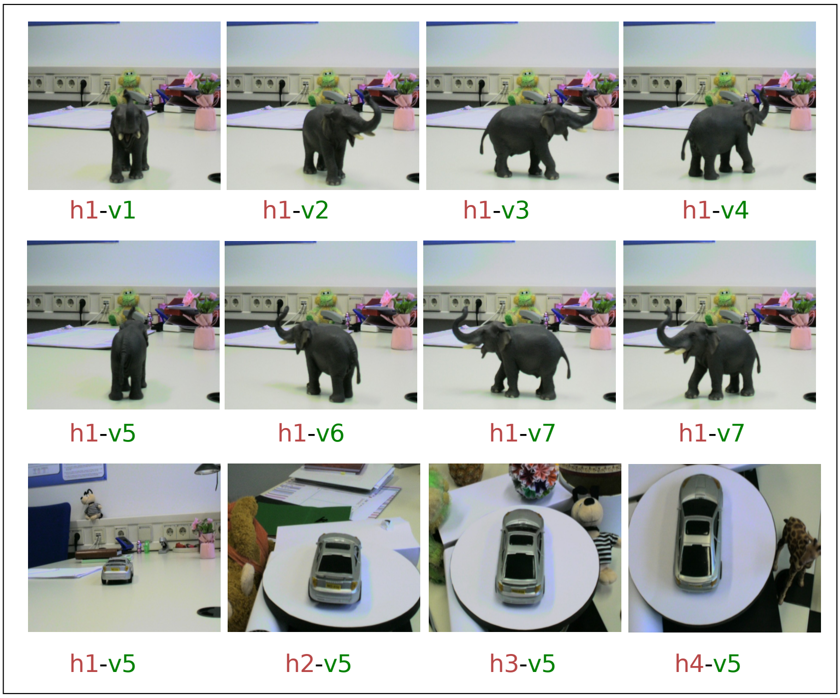


Figure 5.8: Example images of the five object classes of the new object category dataset JenaRange02.


h1-v1 h1-v2 h1-v3 h1-v4

h1-v5 h1-v6 h1-v7 h1-v7

h1-v5 h2-v5 h3-v5 h4-v5

Figure 5.9: Example images of the JenaRange02 dataset at eight different viewpoints as well as the four different heights.

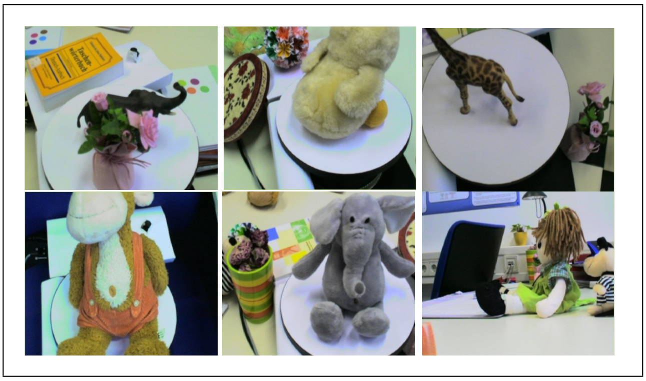


Figure 5.10: Example images of the JenaRange02 dataset with occlusion and truncation.

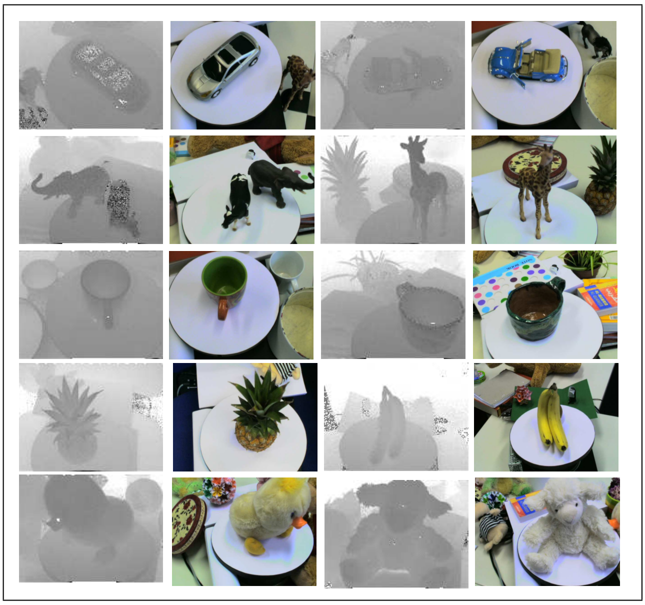


Figure 5.11: *Example range images (using TOF PMD camera) and their corresponding color images (using CCD camera) of the JenaRange02 dataset.* TOF PMD cameras produce reflected images with respect to images produced by CCD cameras.

5.5 Experimental Evaluation

We establish in this section some experiments in order to evaluate the recognition performance using the new dataset and measure how adequate the dataset is for the GOR. To fulfill this goal, we evaluate the JenaRange02 dataset using our generic 2D object recognition model mentioned in chapter 4 using the SIFT-color features combination (SC). Furthermore, we compare the recognition performance using the dataset to the performance of the model using the difficult Graz02 dataset.

For the JenaRange02 dataset, a total number of 300 training images is used, 150 images of the object class and 150 images of the counter-class. A total number of 150 images are used for testing the model. As the dataset does not contain a separate counter-class, a combination of images of other object classes are used as counter-class in training and testing a classifier for an object class. Figure 5.12 shows the instances of each object class used for training and testing the model. Figure 5.13 (second row) displays example images used as counter-class to train and test the class animals.

The recognition results of these experiments are shown in table 5.1 compared to the results of the model using the Graz02 dataset (results displayed in table 4.11). Because each dataset contains different classes with different context, we define an overall recognition performance for each dataset as the average performance over the object classes contained in the dataset. The shown in both tables results reveal the difficult nature of the JenaRange02 dataset compared to a difficult benchmark like Graz02. The high amount of background clutter, viewpoint and size variations as well as occlusions make the recognition using the dataset a difficult task. Moreover, the used counter-class plays an important role here. When a counter-class, which has a completely different context from the context of the images of object classes (positive examples), is used, recognition is then more easier. In contrast, when the context of the counter-class images is almost the same as the images context of the object class, which is the case here, recognition is getting harder as the background of all images (positive and negative) is almost the same. For the recognition model to be able to deliver correct classification results, it must be able to find and account for small differences exist between the images. This explains why Caltech 4 is more easier than Graz02, for example. The following experimental results emphasize this conclusions. We repeated the same previous experiments, but with using different counter-class for each dataset. For, experiments using JenaRange02 dataset, we used the counter-class of the Graz02 dataset (see figure 5.13 last row) and for Graz02, we used the counter-class of JenaRange02 (see figure 5.13 second row). The experimental settings are the same as in the previous

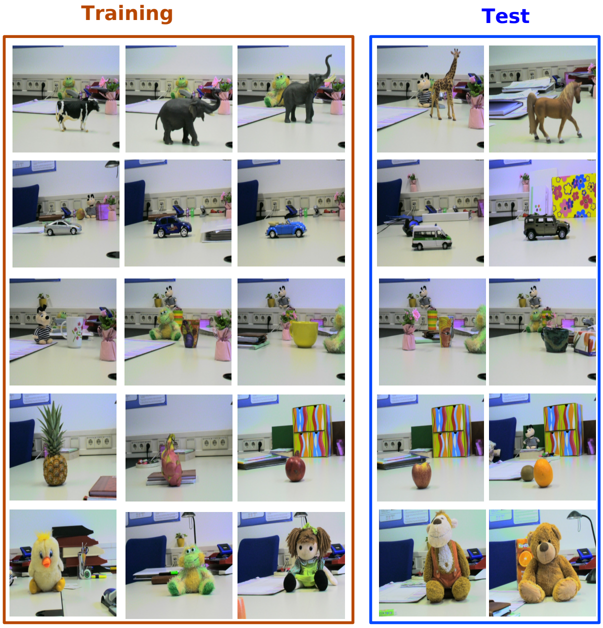


Figure 5.12: Object class instances used to train and test the recognition model.

experiment. The recognition results of these experiments are displayed in table 5.2 and show how easier the recognition is compared to previous experiments. Therefore, the choice of the counter-class is important for investigating the recognition performance of models. The use of a counter-class that has images of the same context as the images of the object class is more realistic and test the robustness of any recognition model.

Table 5.1: Classification rates at ROC-eqq.-err. rates of recognition using JenaRange02 compared to the results using Graz02.

	JenaRange02					Graz02		
Class	Cars	Fruits	Animals	Toys	Cups	Bikes	Cars	Persons
Recog.:	70.80	83.30	70.00	56.90	70.80	80.00	77.33	82.67
Average	70.36					80.00		

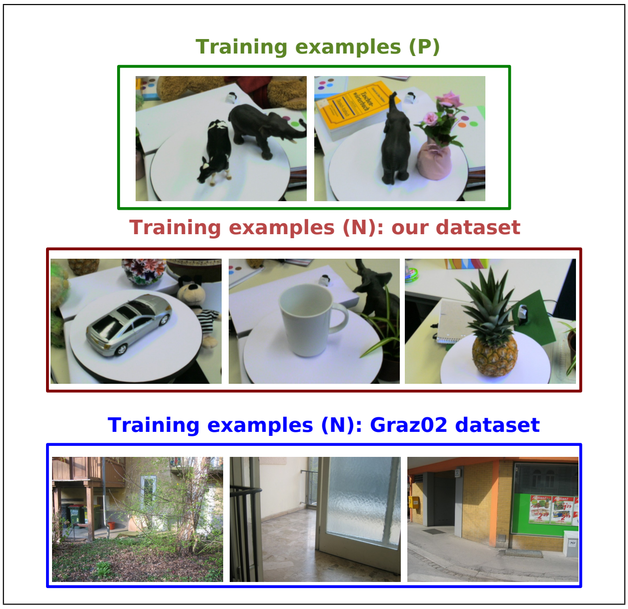

Training examples (P)
Training examples (N): our dataset
Training examples (N): Graz02 dataset

Figure 5.13: Object class instances used to train and test the recognition model.

Table 5.2: Classification rates at ROC-eqq.-err. rates of recognition using JenaRange02 compared to the results using Graz02. Different counter-classes are used.

	JenaRange02					Graz02		
Class	Cars	Fruits	Animals	Toys	Cups	Bikes	Cars	Persons
Recog.:	98.00	98.00	100.00	94.00	98.00	96.00	87.00	90.00
Average	97.60					91.00		

5.6 Conclusions

This chapter has presented our new object category dataset. The dataset has the advantages over all existing datasets of providing range images of its member classes. In addition to the range image, it provides 2D images: grayscale or colored. A Time-of-Flight PMD camera is used in acquiring the range images of the dataset. The dataset has actually two versions. The first one, JenaRange01, provides 2D intensity images as well as range images of instances of three different object classes. The dataset contains inter- and intra-class variabilities as well as large viewpoint variations. However, it provides images with a single object and without background clutter or occlusion (like almost all the available range image dataset. Some are mentioned in section 2.4.3).
The second version of the dataset, JenaRange02, overcomes the limitations exist in JenaRange01 dataset. It provides 2D colored and intensity images, beside the range images. It contains images of five different visual classes with large intra-class variabilities among the classes. Its images contain large degree of size and viewpoint changes as well as background clutter and occlusion. Moreover, the dataset provides context independent images. It is hard to recognize objects from context using this dataset as almost all the images of the different object classes have the same context. Therefore, the use of this dataset offers real performance measure of recognition models. However, JeanRange02 dataset could be used not only for GOR from range images but also form 2D images. Additionally, it could be used for 2D and 3D object recognition tasks as well.
The construction and existence of this dataset would encourage the researchers to exploit range images for GOR. We were the first to tackle this difficult problem by our model, which will be presented in the next chapter.

Chapter 6

A Model for Generic 3D Object Recognition

With the aid or our new dataset which provides range images of different object classes, dealing with GOR from range images is now possible. This chapter addresses the more difficult problem of generic 3D object recognition and presents our model for recognition of generic 3D object classes from range images, which exploits simple local shape features extracted from range images for recognition. As in the previous 2D framework, Boosting is the main learning technique. Experimental evaluations are established showing the reasonable performance of the new recognition model.

6.1 Motivation

One goal of computer vision research is to give computers human-like visual abilities so that machines can sense the environment in their field of view, understand what is being sensed and take an appropriate action as programmed. The existence of a working vision system, the human visual system (HVS), has encouraged the researchers to try to develop an artificial vision system that performs somehow like it. For the machine, the ability to recognize objects in the surrounding environment is very important and since we live in a 3D world, recognizing 3D objects is the most important and difficult part of the object recognition problem in general. This difficulty arises from the infinite viewpoint variations possible for objects in the 3D real-life.

Many approaches addressed the recognition of specific 3D object either from appearance-based information extracted from 2D images such as [12, 107] or from range images with the aid of shape information such as [88, 53]. An overview of some of the approaches is given in chapter 3.

However, the generic recognition of 3D objects is a more challenging extension to the original problem of GOR. Most of the recent researches and approaches in GOR have focused on modeling the appearance and shape variability of objects with limited number of changes in viewing point (e.g. [43, 81, 151]). One main reason is that the most current object category datasets contain images with small variations in viewing point (e.g. Caltech 4 and UIUC cars).

A small number of research has investigated the problem of generic 3D object recognition. One of these approaches is presented by Savarese and Fei-Fei [130] and described in section 3.2.3. Savarese and Fei-Fei [130] also introduced a new 3D object dataset [6] which provide only 2D images of different 3D object classes (refer to section 2.4.1 for more details about the dataset). However, our approach for generic 3D object recognition [62] is totally different from the approach of [130]. The main difference is that range images are used in our proposed approach, which is not the case in [130] as they use 2D images. Furthermore, only surface shape features are used here to represent the instances of the object classes while no appearance information is used as in [130].

Another approaches are presented by Sun *et al.* [138] and Su *et al.* [137] which used also information extracted from 2D images for recognition and made use of the dataset of Savarese and Fei-Fei [130]. Section 3.2.3 present more details about these approaches.

Another approach, which is closer to our work, is described in [129]. The approach developed to recognize objects belonging to a particular shape class in range images as mentioned in section 3.2.3. Although our approach agrees with this one in that surface shape descriptors are used to represent the object classes in real range images, there exist main important differences between the two approaches. First, a combination of three different simple local surface features is used in our approach as a representation of the instance of the different object categories. Second, learning is performed here using boosting which is different from the learning technique, namely Support Vector Machines (SVM), used in [129] . Moreover, a dataset of real range images and of real different object categories is used in our approach (our dataset JenaRange01), while in [129], a dataset of range images of objects made of clay is used. Their dataset is then enlarged by applying deformations to the original clay objects to offer intra-class variabilities . In contrast to [129], our dataset contains large intra-class as well as inter-class variabilities, so it is not necessary to apply any deformation to enlarge it (see figure 6.1).

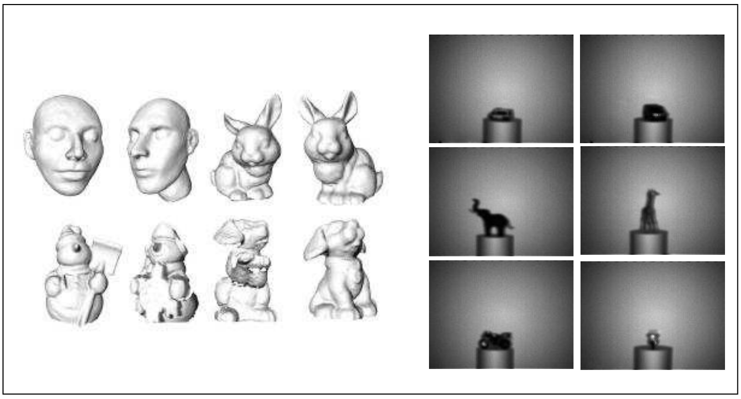


Figure 6.1: *Example images of the dataset used in [129] and our dataset JenaRange01*. The first column(left) shows instances of shape classes mad of clay used in [129]. The second column (right) shows example images of instances of object classes in JenaRange01 dataset.

Addressing the problem of generic 3D object recognition form range images is the main contribution presented in this chapter. Up to our knowledge, no other recognition approaches developed to tackle this difficult problem. However, this chapter presents, additionally, more contributions. The way range images are handled in our recognition model is usually used for 2D images and is different from the way accomplished by all other approaches which used range images for recognition. The recognition framework in general is suited for recognition from 2D images and never been applied for recognition from range images (as far as we know).

Section 6.2 provides an overview of the main steps of the proposed recognition model. Experimental evaluations of the recognition model are presented in section 6.3 with investigating the performance of the model concerning different important aspects of 3D object recognition.

6.2 The Recognition Model

6.2.1 Overview

In this section, the main idea of the proposed generic 3D object recognition model is explained. Figure 6.2 provides a semantic view of the main components of the pro-

posed model. This model consists of three main steps. First, an affine interest point detector is applied to the intensity images to detect a set of interest regions. The detected interest regions are extracted together with their corresponding 3D depth data. Second, simple local surface shape features are computed from the extracted 3D regions. Finally, boosting, namely the RealAdaBoost algorithm [133], is used to learn these simple shape features for each object class. The choice of the RealAdaboost algorithm here is based on the evaluations results presented in section 4.5.
The idea of the proposed model, which is combining and boosting interest point detector together with local descriptors for recognition, is normally used for generic recognition tasks using 2D images and has never been applied to range images. Actually, we adapted our model for generic 2D object recognition proposed in chapter 4 to make it suitable for the new recognition task. This is accomplished by adapting the interest point detection step to make it suitable for the new image type (range images). Moreover, a representation, which meets the characteristics of the range images, is used (shape representation in the form of simple local shape descriptors).

6.2.2 Preprocessing and Interest Regions Detection

Preprocessing: The range data of a TOF chip (in our model PMD) has statistical noise. In order to filter this noise and smooth the range data, a preprocessing step by applying median filter is first performed as in [54]. Furthermore, an initial histogram normalization is applied to the PMD intensity images to enhance their low contrast and improve the interest points detection process.

Interest Regions: An implementation of the Hessian affine-invariant region detector developed by [99] is used to detect and extract interest regions from the 2D intensity images corresponding to range images delivered by the TOF camera. Afterwards, the 3D regions corresponding to the detected points are extracted. Figure 6.3 gives an overview of the point detection procedure.

6.2.3 Local Features Computation

Range images have the advantage of providing direct information about the shape of objects. Therefore, it is wise to make use of this advantage and give preference to features that capture different aspects of this shape. Local shape descriptors (LSD) are preferable as they provide some robustness to clutter and occlusion. There are different LSD features in the literarture (some of them are described in section 2.1.2). The

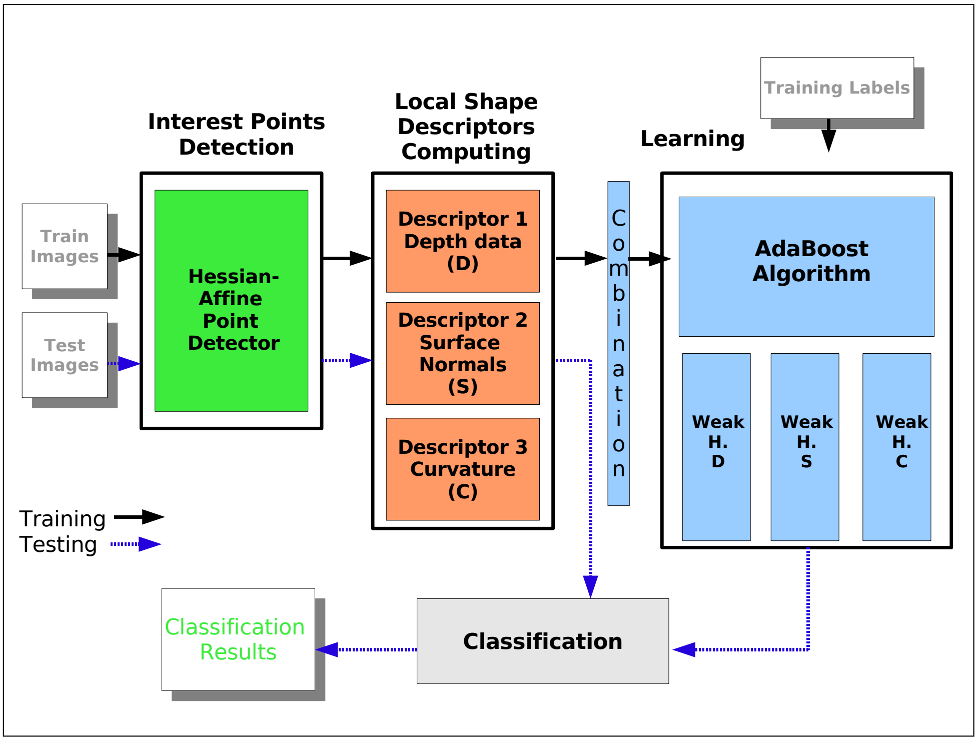

Interest Points Detection
Local Shape Descriptors Computing
Learning
Training Labels
Train Images
Test Images
Hessian-Affine Point Detector
Descriptor 1 Depth data (D)
Descriptor 2 Surface Normals (S)
Descriptor 3 Curvature (C)
Combination
AdaBoost Algorithm
Weak H. D
Weak H. S
Weak H. C
Training
Testing
Classification Results
Classification

Figure 6.2: The proposed generic 3D object recognition model.

most famous one is the Spin Images presented in [70]. Spin Images are, as mentioned in section 2.1.2, a 2D histogram of of the cylindrical coordinates of its surrounding points. Spin Images have shown robustness to occlusion and noise in the experiments reported in [70, 97]. However, long and extensive computation time is required to to obtain accuracy when computing the images. Moreover, the complexity of the images increases with the increase of the size of the dataset. The Point Signatures [32] is another LSD, which is a one dimensional signature that describes the surface surrounding a point. However, Point Signatures descriptor lacks accuracy, is sensitive to the translation of the views and requires high computational complexity. More LSD descriptors are mentioned in section 2.1.2.

Hetzel *et al.* [63] introduced and used three shape-specific local feature histograms for the task of free-form specific 3D object recognition. The features are namely:

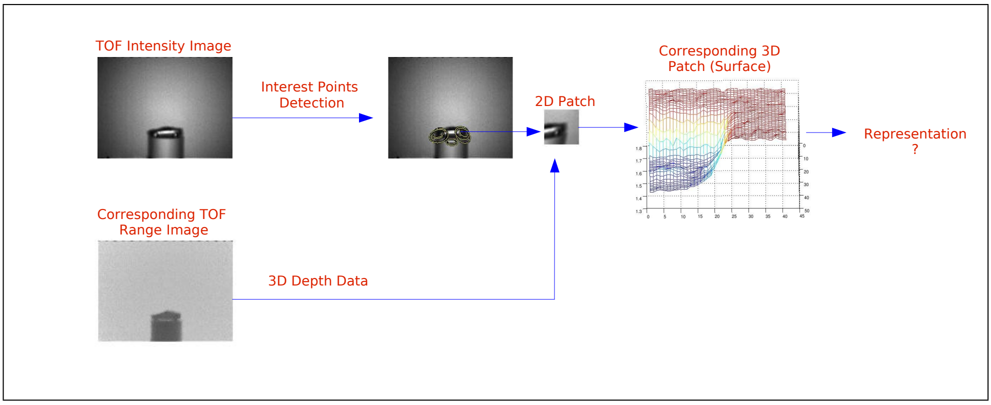

TOF Intensity Image
Interest Points Detection
2D Patch
Corresponding 3D Patch (Surface)
Representation ?
Corresponding TOF Range Image
3D Depth Data

Figure 6.3: The process of interest points detection in the used range images.

pixel depth, surface normals and curvature. The main advantages of these features are that they are easy to calculate, robust to viewpoint changes and contain discriminative information [63]. The mentioned advantages of the features together with the good performance they revealed in [63] for specific 3D object recognition are the reasons to use them and investigate their suitability and robustness for the more difficult problem of generic 3D object recognition. Following, a brief description to these features is given.

Pixel Depth

The distance to the object provided by the PMD camera is the simplest available feature. Computing a histogram of pixel distances provides a simple feature which is invariant against translations and image plane rotations and at the same time gives valuable cues about the shape of the object. However, if there are large and abrupt changes of the depth range, e.g. due to occlusion effects, the whole histogram will be shifted and the recognition might no longer be guaranteed. For this reason, pixel distances histogram can be relied on for the recognition of objects with sufficient depth range [63]. In this work, a histogram of 64 bins of pixel distances is calculated and used.

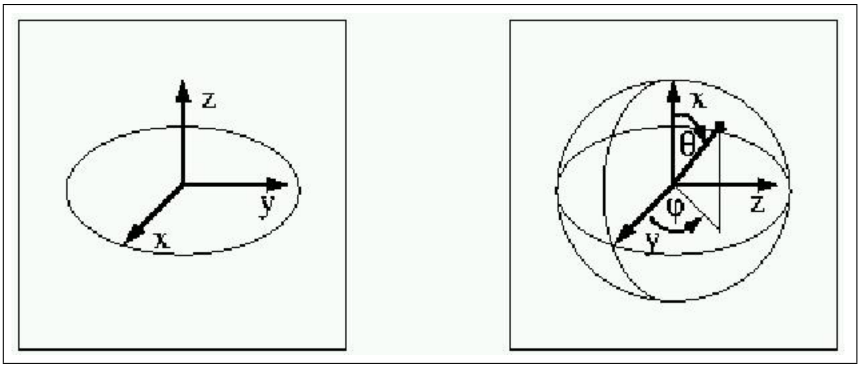


Figure 6.4: Representation of normals in sphere coordinates [63]

Surface Normals

Surface normals can be easily calculated from the first derivative of the image. A representation of surface normals as a pair of two angles (ϕ, θ) in sphere coordinates is presented in [63] (see figure 6.4). This representation is shown to spread over as possible of the available histogram range without having a bias for certain regions [63]. The angles can be calculated as follows:

$$\phi = \arctan(\frac{n_z}{n_y}), \theta = \arctan\frac{\sqrt{(n_y^2 + n_z^2)}}{n_x} \tag{6.1}$$

A two dimensional histogram of size 8 x 8 bins of the of two angles is computed and used.

Curvature

Surface curvatures can be calculated either from the first and second derivatives or as the rate of change of normal orientations in a certain local context region. The pair of Gaussian curvature K and mean curvature H provide only a very poor representation, since the values are strongly related [63]. Instead, the shape index representation introduced in [76] and modified in [36] can be used. The representation is given as follows:

$$S_I = \frac{1}{2} - \frac{1}{\pi} * \arctan\frac{k_{max}(p) + k_{min}(p)}{k_{max}(p) - k_{min}(p)} \tag{6.2}$$

where $k_{max}(p)$ and $k_{min}(p)$ denoting the principle curvatures around the point p. The shape index S_I has the range of $[0, 1]$, and every distinct surface shape corresponds to a unique value S_I (except for planar surfaces, which is mapped to the value 0.5, together with saddle shapes) [63]. The shape index is invariant to translations, but due to limited resolution, it varies with image plane rotations and scale changes [63]. A histogram of shape index of 64 bins is used.

6.2.4 Learning Model

RealAdaBoost algorithm (algorithm 2) is used for learning in our model. The weak learner used here is the same used in section 4.2. However, some modifications are applied to make it cope with combining three descriptors. Moreover, weak hypotheses are selected from positive examples only, while in the weak learner presented in section 4.2 as well as in [113], weak hypotheses are selected from positive and negative examples. Selecting hypotheses from only positive examples has the same (if not better) performance as selecting them from both positive and negative examples [1], and requires less computation time.

6.3 Experimental Evaluations and Results

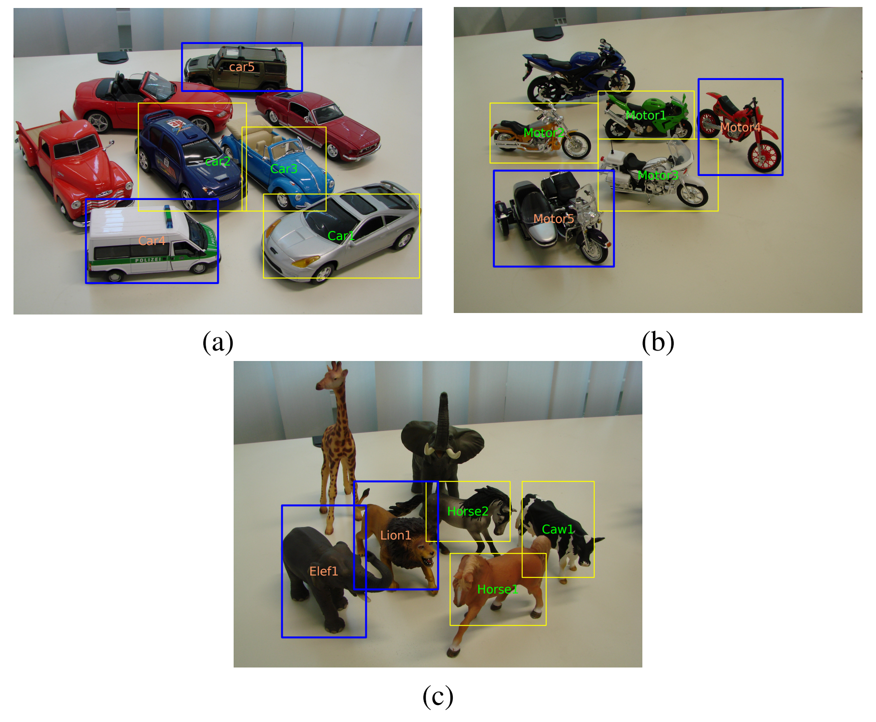
The JenaRange01 dataset is used in all experiments accomplished to investigate the recognition performance of the model. Four sets of experiments are performed to evaluate the performance of the proposed recognition approach with respect to four recognition aspects:

1. The generic recognition ability of the approach.
2. Evaluate the recognition performance with respect to view point change of objects (specific recognition of objects).
3. Test the performance of the generic recognition with viewpoint change.
4. Test the performance of generic recognition in scenes with multiple objects (with the presence of background clutter and occlusion).

RealAdaBoost algorithm is run for $T = 150$ iterations. The model's performance is evaluated using the Receiver-Operating-Characteristic curve (ROC). Moreover, The ROC-Equal-Error rate is computed for each curve.

---

[1] This is experimentally tested.



(a) (b)

(c)

Figure 6.5: *Instances of the object classes of the JenaRange01 dataset used for training and testing the recognition model.* Yellow rectangles indicate the instances of object classes used for training while blue rectangles indicate instances used for testing for each classes.

6.3.1 Experiment 1: Generic Recognition Performance

In this set of experiments, the generic recognition ability of the recognition model is investigated. The object instances used to train the model are different from those used for testing. Figure 6.5 displays the instances used for training and testing for each object class. A total number of 200 images is used for training the model: 100 training images of instances of the object class (positive examples) in addition to 100 training images of the background class (negative examples). A test set of 100 images is used: 50 images of a novel instances of each object class and 50 images of the background class.

Table 6.1: ROC-eqq.-err. rates of the generic performance of the used three object classes (Experiment 1).

Object class	Depth	Normals	Curvature	Combination
Cars	80.00	96.00	80.00	98.00
Motors	94.00	92.00	88.00	98.00
Animals	94.00	92.00	88.00	100.00

Figure 6.6 displays the ROC curves of the recognition results the object classes, while the ROC-Equal-Error rates are presented in table 6.1. The performance of each descriptor used independently for recognition is also displayed in table 6.1 to show the performance gain of combining them together for recognition. The model achieves a high recognition performance on the three used object class. Although the used range images do not contain complex scenes, some difficulties are imposed on the recognition task due to the small size of objects in the images. Detailed variations between different object classes are not clear which makes recognition and classification hard tasks, even for humans (see figure 5.6).

6.3.2 Experiment 2: Recognition Performance with Respect to Viewpoint Change

For a recognition model for 3D objects, recognition of objects from different viewpoints should be robustly achieved. This set of experiments are aimed to test the recognition performance when the viewpoints of objects are different in training and testing. No generic recognition performance is measured in these experiments. This means that, instances of each object class used for training and testing the model are the same. However, the viewpoints of instances used for training are different from those used for testing. Figure 6.7 displays example images used for training and testing the model. The number of training and testing examples is the same as that is used in the previous experiments. The model is able to recognize all test examples of each object class with ROC-Equal-Error rates equals to zero despite the viewpoint change. This means that the proposed model is able to recognize objects independent of the given viewpoint.

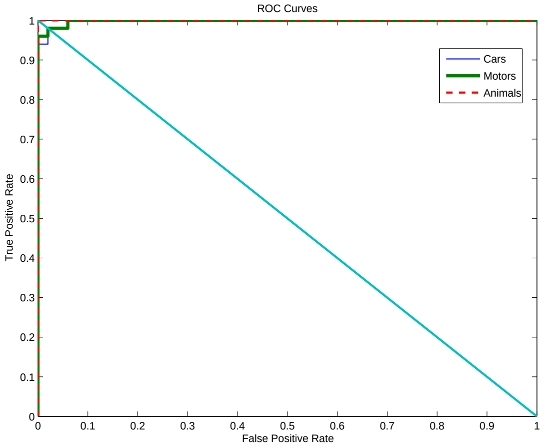
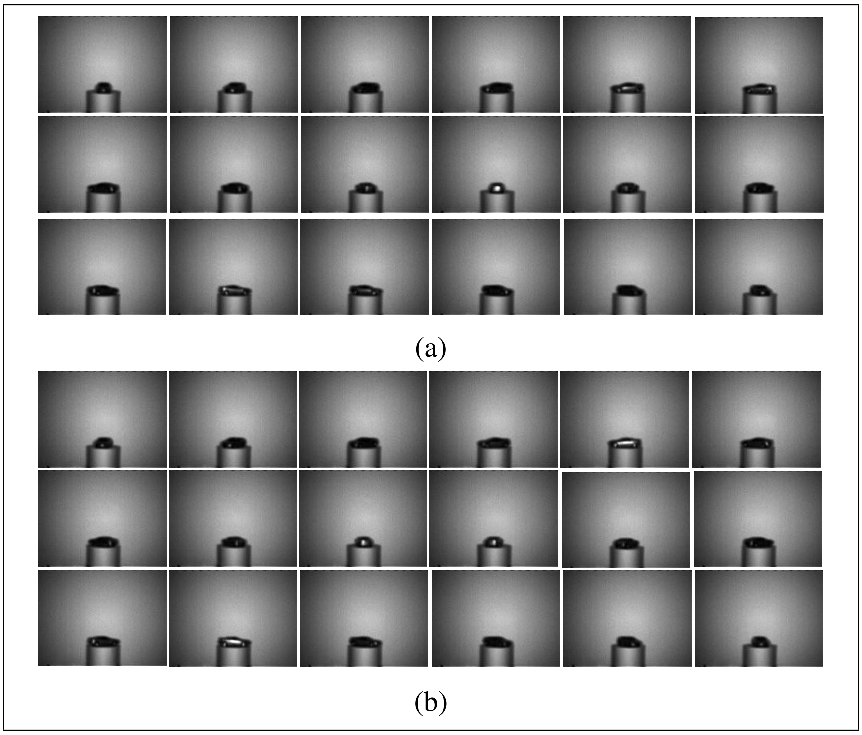


Figure 6.6: The ROC curves of the three classes on the generic recognition task (experiment 1).

6.3.3 Experiment 3: Generic Recognition Performance with Respect to Viewpoint Change

In this experiments, we investigate the generic recognition of the model with respect to viewpoint change. This means that, the instances of each object class used for training are different from those used for testing (as done is experiment 1). Moreover, the viewpoints of training instances are different from the viewpoints of the test instances for each object class. A total number of 100 training examples is used: 50 images of instances of each object class in addition to 50 images of the background class. A total number of 50 test examples are used to test the model. Figure 6.8 displays example of test images used in this set of experiments, while examples of training images are displayed in figure 6.7 (a). The results, which are displayed in table 6.2 shows that the generic recognition performance of the model is robust with respect to the viewpoint change of the objects.



(a)

(b)

Figure 6.7: *Example training and test images for experiments 2.* (a) Example training images of the class cars. (b) Example test images of the class cars. The object instance is the same while the viewpoints are different from those used for training.

6.3.4 Experiment 4: Generic Recognition in Scenes with Multiple Objects

Almost all the approaches that depended on range images for recognition used images with only a single object and without any background and occlusion (*e.g.* [88, 53, 63, 129]) . This is due to the reason that all range image datasets provide images with only a single objects in the scene. However, in this set of experiments we aim to evaluate the effect of background clutter and occlusion on the recognition from range images. Of course, degradation in recognition performance is expected. However, the degree of this degradation is to be measured.

For accomplishing this task, a new set of test images for each object class is recorded

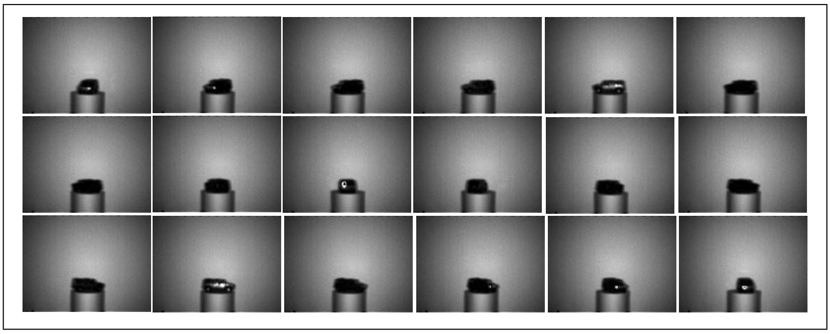


Figure 6.8: Example test images used in experiment 3.

Table 6.2: ROC-eqq.-err. rates of the generic recognition performance of the used three object classes with respect to viewpoint change (experiment 3.

Object class	ROC-equal-error
Cars	98.00
Motors	96.00
Animals	100.00

(see figure 6.9) . These new test images contain occlusion and background clutter by placing instances of each object class (different from the instances used in training) together with instances of new previously unused object classes. A total number of 36 range images from different view points are then recorded for each object class. No new training is performed in this set of experiments. The trained model of section 6.2.3, where the used images are with only single object, is tested using the new test images. A total number of 100 test images is used: 50 positive images and 50 negative images. The ROC curves of the recognition results are shown in figure 6.10 and the ROC-Equal-Error rates are displayed in table 6.3.

Obviously, the recognition performance in these experiments degrades, compared to the previous experiments due to the presence of occlusion and clutter. However, high deterioration in performance did not occur. Beside that, the low resolution of the intensity images of the PMD camera affects the detection performance of the point detector, which influences in turn the categorization performance. Figure 6.11 displays examples of incorrectly classified images (false negatives). We can notice from the figure

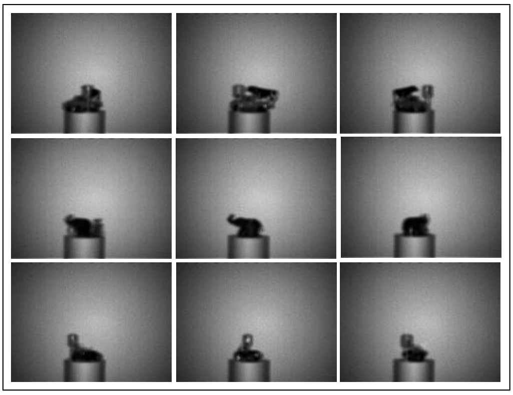


Figure 6.9: Example of the images recorded for the task of recognition in scenes with multiple objects.

Table 6.3: ROC-eqq.-err. rates of recognition using complex scenes for the used three object classes.

Object class	ROC-equal-error
Cars	82.00
Motors	84.00
Animals	80.00

that it is difficult to identify the different objects in the images, which gives more insight on how difficult the problem is, even for humans.
Another important aspect concerning the recognition model is the computational time needed for the training the testing processes. The average training time of the model is approximately 26 minutes for each object class while the test time for a whole test set is approximately one minute for each class.

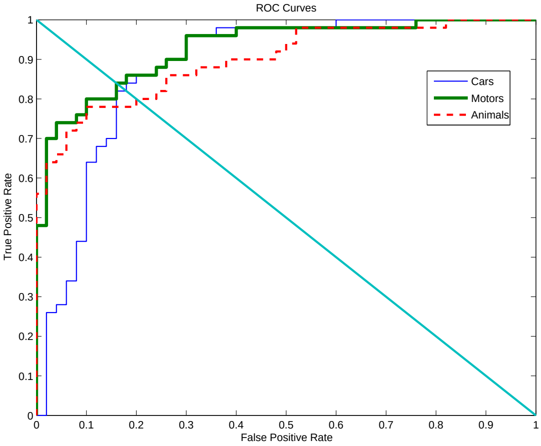


Figure 6.10: The ROC curves of the three classes on the categorization with the presence of clutter and occlusion task.

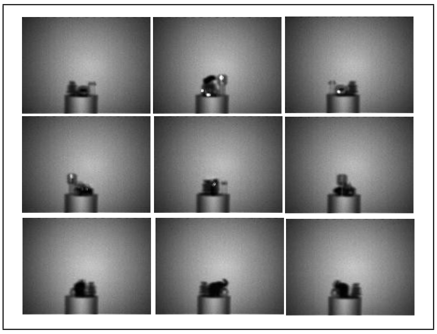


Figure 6.11: *Examples of incorrectly classified images (false negatives) of experiment 4.* Images form the class cars (first row), motors (second row) and animals (third row) which are classified as to belong to the background class.

6.4 Conclusions

In this chapter, we have presented a novel model for generic recognition of 3D objects from range images. This model is considered to be the first one to tackle this difficult problem. The main framework of the model consist of three main steps: interest point detection in range images, features extraction and learning. A combination of three simple local shape descriptors has been used for recognition and has shown to be profitable. Boosting was the underlaying learning technique for binary learning and classification. Our dataset, JenaRange01, has been used for the learning and evaluation processes in the model.

The general recognition performance of the model was promising. The model has shown robust performance for the generic recognition of the different objects instances with different viewpoints. Moreover, the recognition using images with multiple objects was satisfactory. However, the small size of the objects within the used images imposed difficulties in recognition . Moreover, the low resolution of the intensity images of the TOF PMD cameras had some effect on the correct detection of interest points, especially in the experiments with multiple objects in the images. One solution to this problem is to use a high resolution intensity images by combining the PMD camera with a 2D camera of high resolution images (*e.g.* [122]). Another solution is to combine different information cues, such as appearance information, with the range-based information for improving the recognition performance.

The model has shown good performance for the binary classification task. However, multi-learning and classification of 3D objects is more difficult, but realistic mission which should be investigated.

Chapter 7

Multi-Class Recognition

In this chapter, we extend our generic 3D recognition model presented in chapter 6 to cope with the task of recognizing many different object categories. An overview of the model is given, with focusing on the multi-class learning step which is accomplished with boosting. Experimental evaluations of the multi-class recognition model are performed and improvement suggestions are presented, experimented and discussed.

7.1 Motivation

The previous chapter has presented a model for generic 3D object recognition from range images. The model has revealed good performance for the binary learning and classification task. However, an important goal for the machine vision in general is to build a system which is able to recognize many different object categories in a cluttered world. Although the main problem of generic object recognition remains unsolved, some progress has been made on restricted versions of this goal (multi-class learning and classification) [1]

There are different GOR models developed for multi-class learning and recognition such as [144, 115, 130]. Some of these approaches are mentioned in section 3.2.4. Our model [61] differs from the models presented in [144, 115] in that, these models address the problem of generic 2D object recognition, where our model addresses the problem of generic 3D object recognition. Moreover, information extracted from 2D

---

[1] This is for the case of generic 2D object recognition. For generic 3D object recognition problem from range images, this is the first model (up to our knowledge).

images, either appearance-based [144] or shape-based [115], is used by these models. The multi-class recognition approaches presented in [130, 131, 138, 137] perform generic 3D object recognition like in our model. However, they do not use any range-based information in their recognition.
The main contribution presented in this chapter is introducing a model for multi-class learning and recognition of 3D objects from range images. Section 7.2 presents the main steps of the proposed multi-class recognition model. Experimental evaluations are performed in section 7.3 to asses the recognition (categorization) performance of the model. Improvements to the model are suggested and explained in section 7.4 by using grid samplingfor range images.

7.2 Multi-Class Recognition Model

7.2.1 Model Overview

The main framework of the model is almost the same as the one presented in 6.2. However, since the model presented here aims to perform multi-class learning and recognition of generic classes of 3D objects instead of just performing binary classification, the learning step is different from the one presented in 6.2. A Boosting algorithm for multi-classification task is used. All other main steps of the model remain the same as in the recognition model described in section 6.2 (see figure 6.2).

7.2.2 Learning Model

As previously mentioned, the recognition model performs multi-class learning and classification of object categories. In this case, a set of N labeled training examples (I_i, l_i) for $i = 1 \ldots N$ of C object classes are given, where the instances I_i are in some domain χ and the labels $l_i \in 1, .., C$. The task is to decide the class category $c \in \{1, .., C\}$ of an object presented in a new test image. This recognition task is also called categorization
Boosting is the underlaying learning technique. Authors in [144] presented a boosting algorithm, namely the Joint Boosting algorithm, for multiple objects classification which depends on training multiple binary classifiers at the same time and sharing features among them. The algorithm has the advantage that less training data is needed since many classes can share similar features. Learning in our model is based on the Joint boosting algorithm. In contrast to [144], in our model, combined features are

shared among the classes instead of sharing a single feature. This is done through the weak learner used by our learning model, which is different from the one used in [144].

Joint Boosting algorithm

The idea of the Joint boosting algorithm is presented in section 2.2.1. We adapted the JointBoosting algorithm to cope with our different features and with learning shared features instead of learning single feature. Algorithm 7 gives a summarization to the modified version of algorithm. The weak learner used in our learning model is the one mentioned and used in 6.2.4. However, some modifications are applied to it to make it support feature sharing among classes.
Instead of exploring all $2^C - 1$ possible subsets S_n of the jointly trained classes C, we employ the maximal greedy strategy described in [144] where authors have shown that this approximation does not reduce performance dramatically. This starts with the first class that achieves alone the best error reduction. Then we select the second class which has the best error reduction jointly with the first class. We keep adding classes till all classes have been added. At the end, the set from the considered C with the largest error reduction is chosen.
In the recognition step, for each classes subset S_n, we sum the output of the weak learners associated with this subset to get a strong learner for S_n dented $G^{S_n}(I_t) = \sum_{m=1}^{Tn} h_m^n$ (I_t is test example). Afterward, for each class c, we find all subsets S_n that contain c and sum their strong learners to compute the final strong classifier $H(I_t, c)$.

Shared Weak Hypothesis

For the classes $c \in S_n$, a weak hypothesis for each class is computed. Then, the weak hypothesis which classifies the set of classes S_n against other classes (*i.e.* $c \notin S_n$) with the lowest error is chosen to be shared among this set of classes S_n. For computing the weak hypotheses, the weak learner mentioned in section 6.2.4 is used.

7.3 Experimental Evaluation

7.3.1 Experimental setup

The presented recognition model is evaluated experimentally to analyze its benefits and limitations. The performance is measured in three cases:

```
1) Initialize the weights $w_i^c = 1$ and set
$H(I_i, c) = 0, i = 1 \dots N, c = 1 \dots C$.
2) Repeat for $k = 1, 2, \dots T$
    a) Repeat for $n = 1, 2, \dots 2^C - 1$
        i) Train weak learner using distribution $\mathbf{w}^n$ and returns
           a weak hypothesis $h_k^n(c)$.
        ii) Evaluate the error
            $J_{wse}(n) = \sum_{c=1}^{C} \sum_{i=1}^{N} w_i^c (z_i^c - h_k^n(I_i, c))^2$
.
    b) Find best subset: $n^* = argmin_n J_{wse}(n)$.
    c) Update the class estimates:
       $H(I_i, c) := H(I_i, c) + h_k^{n^*}(I_i, c)$.
    d) Update the weights: $w_i^c := w_i^c e^{-z_i^c h_k^{n^*}(I_i, c)}$.
Output: a strong hypotheses $H(I_t, c) = \sum_{o=1}^{O} G_o^{S(n)}(I_t)$
where $O$ is number of subsets $n : c \in S$ and
$G^{S(n)}(I_t) = \sum_{m=1}^{Tn} h_m^n$.
```

Algorithm 7: Modified Joint boosting algorithm [144]. I_i is the i^{th} training example,I_t is test example $z_i^c \in \{-1, +1\}$ are labels of class c and w_i^c are the unnormalized example weights. N is the number of examples and T is the number of boosting iterations.

1. Classes are independently learned. A binary classifier of each class against other classes is run independently.

2. Classes are learned jointly without feature sharing. This means that, in algorithm 7 the given classes C are investigated instead of investigating subsets of the classes. Accordingly, step (a) in algorithm 7 is changed to $n = 1, 2, \dots C$. In this case, a weak hypothesis to distinguish this class only from other classes is to be found. The class, in which weak hypothesis achieves the best error reduction,

is then chosen and the weights distribution of this class is updated.

3. Classes are learned jointly with feature sharing as shown algorithm 7.

In all experiments, the number of training iterations (number of weak hypotheses) is fixed to $T = 150$ and is independent of the number of classes. In contrast to the learning model in [144], we are not searching and comparing the learning effort for a certain error rate, but we report the ROC-Equal-Error rate for a certain learning effort, namely T weak hypotheses. All experiments are performed using our new object category dataset JenaRange02, which is a difficult one as shown in 5.6. However, recognizing the object classes of this dataset using only range information is expected be more difficult than the recognition using the appearance information. We use the five classes: cars, fruits, animals, toys and cups. Figure 7.1 shows the instances of each class used for training and testing the model. The number of training examples for each class is 100 examples, which results in a total of 500 training examples. For testing, 60 examples per class (images of new instances) are used (a total of 300 test examples).

7.3.2 Results

Table 7.1 displays the categorization performance of the model over the used object classes. As expected, categorization of the dataset using only range information is a hard task, and the task is getting harder when multi-class recognition is aimed. The difficulty of multi-class recognition from range images is due to different reasons:

1. The dataset is in general a difficult one, with high percentage of background clutter and occlusion. The context of images of the different object classes is almost the same, which make categorization more harder [2].

2. Another reason is the low resolution of the intensity images of the TOF camera, which are used for point detection in range images. This low resolution of the images (which are, additionally, full of background clutter) affects the detection performance of the point detector and influences, in turn, the classification performance.

3. Moreover, the noisy nature of the TOF range images affects the construction of a clear shape representation for each class, which has an effect on the recognition performance.

---

[2] Results provided in 5.6 emphasize this.

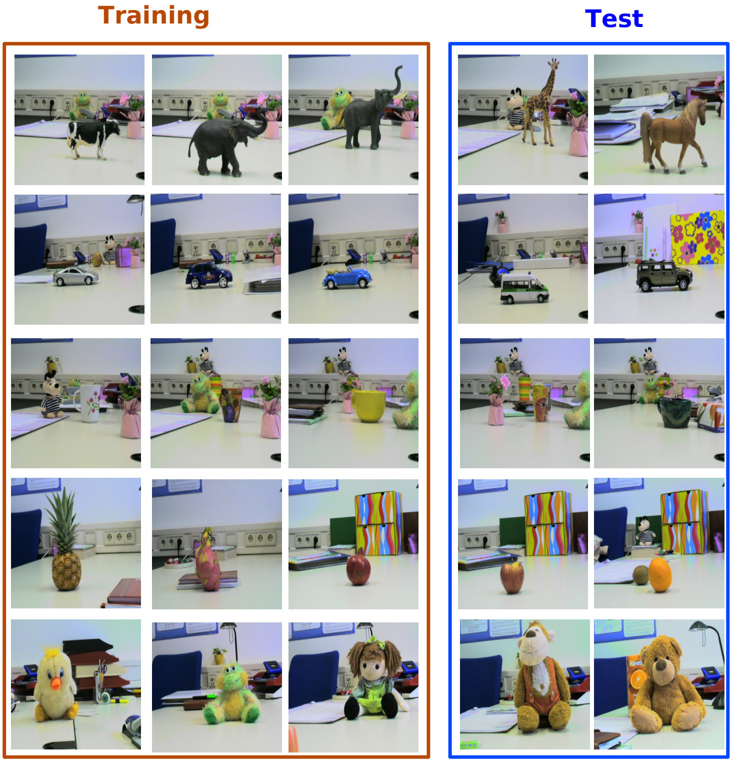


Figure 7.1: Instances of the different object classes used to train and test the recognition model.

Figure 7.2 shows examples of incorrectly classified images from each object class, where columns represent true classes and rows represent predicted classes. We can notice from the figure that the failure in classify some images is due to the existence of an instance (or instances) of the predicted object class in the images such as the image of animals (figure 7.2, second row). It is classified as containing fruits which is not wrong as it actually contains an instance of the class fruits. Another examples are the animals image (figure 7.2, fourth row) and the cars image (figure 7.2, last row)). However, the previously mentioned reasons 2 and 3 are behind the classification failure in the other images.

Table 7.1: *Classification performance at the ROC-Equal-Error rates of the multi-class recognition using range-based information.* The model is either trained jointly with feature sharing (Joint (Sharing)), jointly without feature sharing between the classes (Joint (No sharing)) or independently (Independent). The last column represent the average classification rates over the five classes.

Descriptors	Cars	Fruits	Animals	Toys	Cups	Avg. recog. over classes
Joint (Sharing)	60.00	45.00	56.00	74.00	65.00	60.00
Joint (No Sharing)	58.30	50.00	68.30	81.70	63.70	64.40
Independent	66.00	57.00	75.00	76.00	68.40	68.50

Joint vs. Independent Learning: Results is table 7.1 show that sharing features among the different object class does not significantly improve the categorization performance of all the object classes as expected. Joint learning without feature sharing achieves in contrast better performance while independent learning of the class achieves the best performance over almost all classes. This is also clear from table 7.2, where the confusion matrices of the categorization with each different learning procedure is presented. In the case of joint learning with feature sharing, there is high amount of confusions between the classes. This confusion is reduced when learning is performed jointly but without sharing features. However, independent learning yields less amount of confusion than the other two learning procedures.

The use of point detection in the TOF intensity images is not done in a proper way as we mentioned, and this affects in turn the categorization performance. Therefore, another region sampling method for range images is required for improving the categorization performance. Next section presents a method for sampling range images on a dense regular grid, which will achieve some performance improvement.

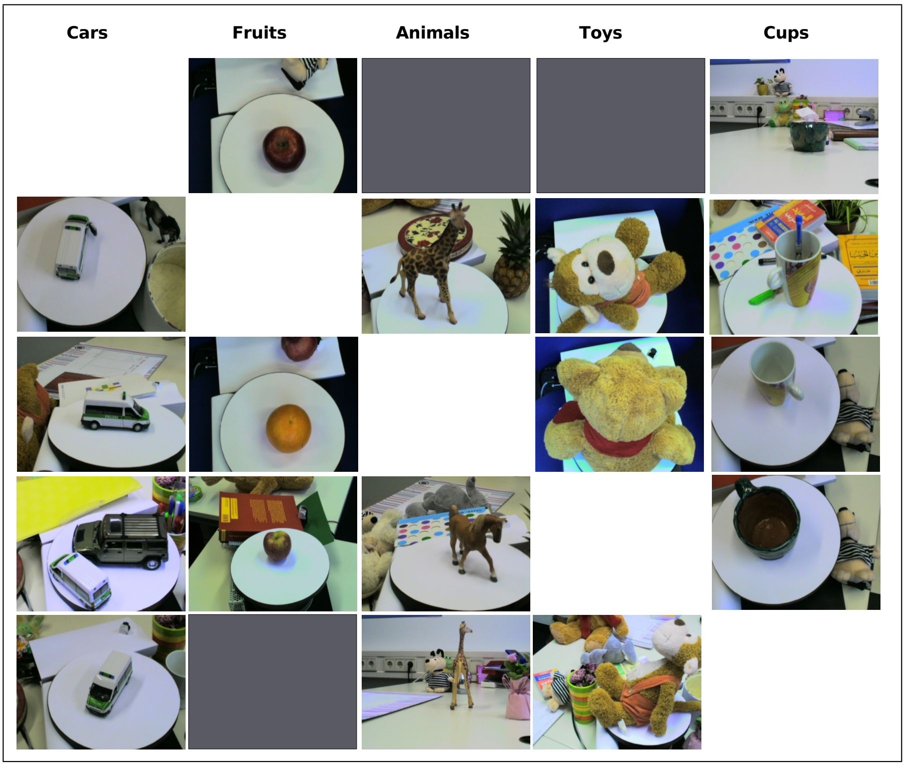


Figure 7.2: *Examples of incorrectly classified images from JenaRange02 dataset using range-based information*. Columns represent true (actual) classes while rows represent predicted classes: cars (first row), fruits (second row), animals (third row), toys (fourth row) and cups (last row).

Table 7.2: *Confusion matrices of multi-class recognition using range-based information.* Comparison of results using joint learning with feature sharing (Jointly (Sharing)), joint learning without feature sharing (Jointly (No Sharing)) and independent learning (Independently) respectively. For the computation of the confusion matrix, the best classification of an image over all classes is counted as the object category. Numbers represent percentage (%) of test images (60 images per class) classified for each class. Columns represent true classes.

	Jointly (Sharing)					Jointly (No Sharing)					Independently				
Class	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
Cars:c1	24	6	14	3	**19**	**70**	**82**	32	25	**65**	**48**	18	0	0	27
Fruits:c2	0	11	8	6	0	2	17	1	8	2	12	**47**	10	10	5
Animal:c3	**28**	**39**	**25**	14	**19**	18	1	**37**	5	18	20	2	**49**	3	16
Toys:c4	2	4	7	**35**	4	5	0	11	**60**	0	15	33	28	**85**	20
Cups:c5	6	0	6	2	18	5	0	10	2	15	5	0	13	2	**32**

7.4 Dense Grid Sampling of Range Images

As we mentioned before, intensity images delivered by TOF camera are of low resolution, which affects the direct application of some images processing techniques such as interest point detection, and which affects in turn the recognition processing. Although our model of recognition from range images, which depends on point detection for range image representation as mentioned in section 7.2.1, achieves reasonably good performance (taking into consideration the difficulty of the original problem) performance improvement is needed. Finding another way to sample local regions for range images can improve the performance. One option is the use of a suitable 3D point detection algorithm, where a set of 3D salient points are to be detected in the range images and the corresponding 3D regions are extracted and described using a suitable description method. This sampling method (3D point detection) used by different approaches for specific 3D object recognition such as [88]. However, the noisy nature of the TOF range images is expected to prohibit the 3D point detection algorithm from working in the prober way and hence delivering the expected performance.

Some approaches for object and scene recognition such as [104] do not depend on point detection methods for locally sampling the images. This is for two reasons: 1) keypoint detectors are considered sometimes to be not repeatable enough in case of object categories which have large intra-class variabilities [104] and 2) they do not always detect all important regions and information in images which can help for delivering better performance. These approaches used different methods for local region sampling such as on dense grid. In sampling local regions form images on a regular dense grid, the 2D images are divided into dense grid of uniformly spaced cells (patches) on which the local descriptors are afterwards computed as in [26]. We follow the approach of [26] and use the dense grid representation for locally sampling the images. However, we differ from the approach in [26] in that they apply this sampling method on 2D images, while we use it for sampling range images.

Now, range images in our recognition model are divided into square patches of size $N \times N$. The patches are spaced by M size on a regular grid. The patches do not overlap when $N = M$ and do overlap when $N > M$. Figure 7.3 shows an example of the grid sampling of range images, while figure 7.4 gives an example of a range image with grid sampling at different values of the parameters N and M.

After the process of grid computation, the three different local shape descriptors are computed from the resultant 3D grid patches and the learning process goes further in the same manner as the procedure described previously in 7.2.1.

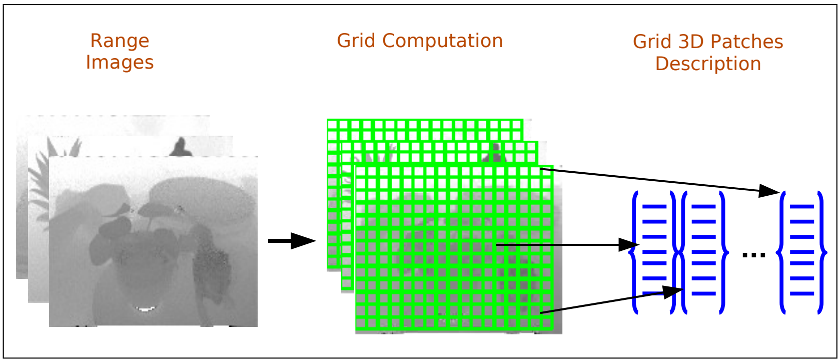


Figure 7.3: Example of range images description using dense grid.

7.4.1 Experiments and Results

Now, we evaluate the performance of our recognition model for generic 3D object recognition with the dense grid sampling of range images. The experimental setup of the following experiments are the same as described in 7.3.1.

The parameters N and M are important for the dense grid sampling as mentioned in [26]. Therefore, the first set of experiments is aimed to asses the effect of the values of both parameters on recognition performance of the model. Learning the different object classes in this set of experiments is performed independently. Figure 7.5 (a) displays the model's performance at different values of the patch size $N = \{5, 7, 9, 11, 13, 15, 20\}$. The value of the spacing size M is set in these experiments equal to N, which means that there is no overlap between the grid patches. The performance is measured as the average classification rates at ROC-Equal-Error rate of the used object classes. We can notice from the figure that the performance increases progressively with the increasing of the value of N until $N = 15$ and then slightly drops off. To asses the effect of the parameter M, which controls the spacing between grid patches and hence controls if there is overlapping or not among them, we repeated the experiments at two different values of $N = \{15, 20\}$, namely the values where the best recognition performance is archived in figure 7.5 (a). The values of M are varied to asses the performance at two cases: overlapping ($M < N$) and non-overlapping ($M = N$) among the grid patches. Figure 7.5 (b) displays the results of these experiments. At $N = 15$, allowing overlapping among the grid patches achieves almost the same results as without overlapping, while allowing overlapping among the

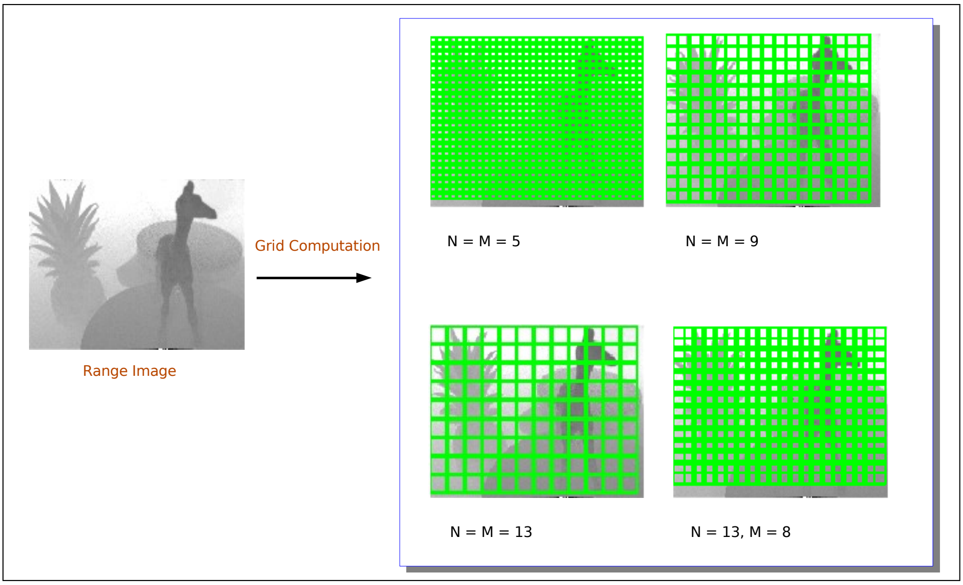
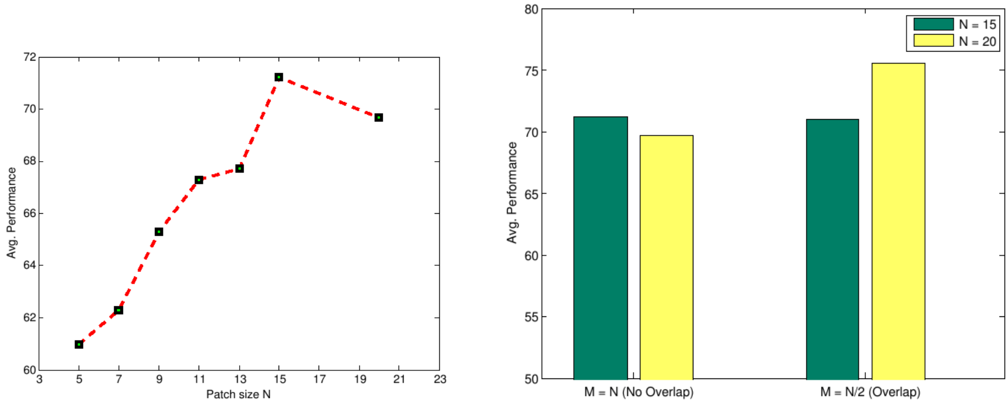


Figure 7.4: *Dense grid sampling for range images*: show grid computed with different values of patch size N and spacing size between patches M. No overlap between patches when $N = M$. When $N > M$ *e.g.* $N = 13$ and $M = 8$, patches are overlapped.

patches achieves better over all performance (75.6%) at $N = 20$.
All previous experiments are performed with independent learning of the object classes. We repeated the experiments with learning the classes jointly with feature sharing and jointly without feature sharing respectively. The value of N is set to 20×20 and with overlapping among the grid patches ($M = \frac{N}{2}$) in all experiments.
The recognition results in the form of classification rates at the ROC-Equal-Error rate using the test set are displayed in table 7.4 while table 7.3 displays the confusion tables of recognition using the test data. Again, independent learning of the object classes achieves better performance than the joint learning in general.
Figure 7.6 displays a comparison of the model performance using the range image sampling with the two different methods: point detection and dense grid. It can be noticed from the figure that the use of grid sampling improves the recognition performance over almost all classes for all different learning procedures. However, for the classes



(a) (b)

Figure 7.5: *Effect of the parameters N and M on the recognition performance of the model*: (a) Effect of patch size N with no patches overlap ($N = M$). (b) Effect of patch spacing size M with M set to values allowing overlapping and non-overlapping between the grid patches. The values of N are chosen from results displayed in (a) where the best two recognition rates are achieved.

animals and cups, the use of point detection for range images sampling achieves better performance than using dense grid.

Table 7.3: *Confusion matrices of multi-class recognition using range-based information with dense grid sampling of range images*: Comparison of results using joint learning with feature sharing (Jointly (Sharing)), joint learning without feature sharing (Jointly (No Sharing)) and independent learning (Independently) respectively. Grid representation of range images is used with patches size $N = 20 \times 20$ and patches overlap ($M = \frac{N}{2}$). For the computation of the confusion matrix, the best classification of an image over all classes is counted as the object category. Numbers represent percentage (%) of test images (60 images per class) classified for each class. Columns represent true classes.

	Jointly (Sharing)					Jointly (No Sharing)					Independently				
Class	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
Cars:c1	32	15	17	10	33	18	7	5	2	3	**48**	0	15	5	13
Fruits:c2	10	**70**	7	13	4	2	**57**	8	17	2	6	**92**	8	8	3
Animal:c3	22	5	33	7	28	37	6	32	5	30	28	7	**59**	12	32
Toys:c4	0	0	2	**52**	0	0	2	0	**68**	3	0	0	8	**70**	7
Cups:c5	**36**	10	**41**	18	**35**	**43**	28	**55**	8	**62**	17	2	10	5	**45**

Table 7.4: *Classification performance at the ROC-Equal-Error rates of the multi-class recognition using grid representation of range images.* The dense grid patches are of size $N = 20 \times 20$ with patches overlap ($M = \frac{N}{2}$). The model is either trained jointly with feature sharing (Joint (Sharing)), jointly without feature sharing between the classes (Joint (No sharing)) or independently (Independent). The last column represent the average classification rates over the five classes.

Descriptors	Cars	Fruits	Animals	Toys	Cups	Avg. recog. over classes
Joint (Sharing)	58.33	75.00	57.00	93.00	56.70	68.00
Joint (No Sharing)	73.30	73.30	61.70	93.30	66.30	73.52
Independent	73.00	90.67	55.00	91.67	66.70	75.41

7.5 Conclusions

This chapter has presented an idea for extending our model for generic 3D object recognition from range images introduced in chapter 6 to the the multi-class learning and recognition case. Based of the idea of the Joint Boosting algorithm of [144], we explored how range-based information can be shared among different object classes. The experimental evaluations of the recognition model have revealed promising categorization performance. However, feature sharing among different classes seems to be not suitable for learning in our case, as it does not show significant performance gain. This could be due to different reasons. Either the features used for recognition do not support feature sharing in a proper way or the inter-class variabilities among the different object classes is high in that sharing features among them is not useful.

Suggestion for performance improvement by using dense regular grid sampling for range images has also been proposed, experimented and evaluated. However, this way of locally describing range images is never been applied before. The use of this sampling method has resulted in notable categorization performance gain.

In general, we think that the performance of the proposed generic 3D recognition from range images model is good. However, there are different drawbacks of the model which need to be improved. These drawback appear more in the multi-class recognition case. One of these drawbacks is that the used descriptors are simple and do not help to form a good shape representation for each object class. This is clear from the amount of confusion among the different object classes. Of course, part of this confusion is also due the noisy nature of the used range images in general. However, the use of more robust shape descriptors would improve the performance.

Since our model is considered to be the first one in its area, there is no room of comparisons available to to be able to further asses its performance. However, the existence of our new datasets as well as our models will encourage the researchers to tackle this problem. In this time, assessing the recognition performance through comparisons would be possible.

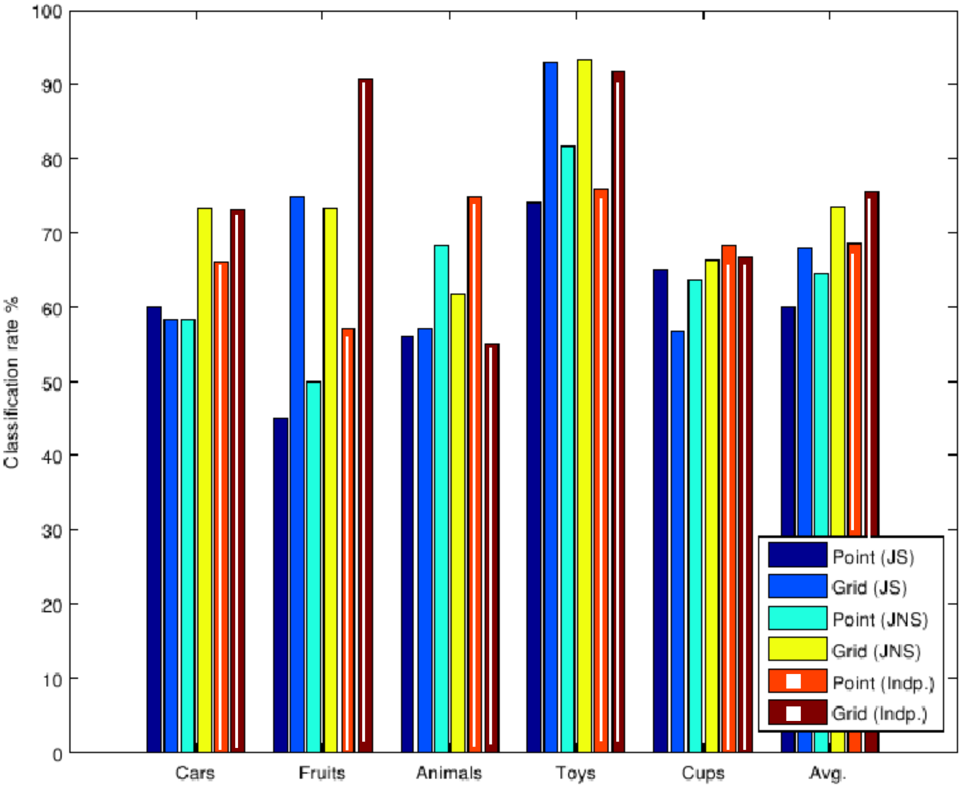


Figure 7.6: *Comparison of multi-classification results between the range images sampled using points detection (Point) and dense grid (Grid) respectively*. Learning is performed either jointly with feature sharing (JS), jointly without feature sharing (JNS)) and independently (Indp) respectively.

Chapter 8

Fusion of Appearance and Range Information

In this chapter, we investigate how different information cues could be combined for generic 3D object recognition. We present a model which exploits a fusion of appearance-based information extracted from 2D images and range-based information extracted from range images for multi-class generic 3D object recognition. First, we show how these different information cues are combined. Then, we investigate and evaluate the categorization performance achieved by this information fusion.

8.1 Motivations

In chapters 6 and 7, we have presented our novel model for generic 3D object recognition from range images which has revealed promising performance in both binary and multi-class recognition cases. However, performance improvements are still required and needed. The model is solely dependent on simple representations of objects shape extracted from range images. However, as already shown in [162, 111, 153], the use of complementary features lead to performance gain. Hence, we combine different complementary features which represent appearance and shape information of the different object classes for recognition and categorization.

Many different approaches used a combination of different information cues for GOR. Opelt [111] presented a hybrid recognition model that combines region based appearance information with shape information based on their boundary fragments model (BFM). The used different information are extracted from 2D images. They also used boosting, namely the AdaBoost algorithm, for learning a binary classifier for each class. Zhange *et al.* [162] combined local texture features, (PCA-SIFT), global features (shape context) and spatial features extracted from 2D images within a single multi-layer AdaBoost model for object class recognition. Moreover, Viola *et al.* [153] combined appearance and motion information extracted from videos for pedestrian recognition.

Our recognition model [61] differs from all other approaches in that, a combination of range-based information and appearance based 2D information is used for generic 3D object recognition, while most of the approaches make use of a combination of complementary information extracted from only one information source (*e.g.* 2D images as in [111] or videos as in [153]). The combination of these different information cues extracted from different information sources (2D and range images) is the first contribution presented by our model. Such a combination has never been used before (up to our knowledge) for GOR. Exploiting this combination for multi-class generic 3D object recognition is the second contribution of our model.

The general aim of this chapter is to investigate how much such a combination can be done and what gain it achieves in categorization performance. Section 8.2 provides a description of the general framework of the recognition model and presents how the different information cues are combined. Experimental evaluations are performed and presented in section 8.3, which reveal the benefits and drawbacks of the proposed information fusion.

8.2 Model Overview

Figure 8.1 displays the framework of the general idea of the fusion of appearance and range information. Actually, the model merges our appearance based model described in chapter 4 and our range-based model described in chapters 6 and 7 into one model for multi-class generic 2D as well as 3D object recognition.

In the model, two different image types are presented: 2D images and corresponding 3D range images. A set of different local descriptors are extracted from the images based on the image type. These local descriptors are computed from interest regions detected and extracted form the images (also according to the type of the image). Then, a combination of the different local descriptors is given to a boosting learning algorithm, namely the JointBoosting, for multi-class learning. Recognition of new test images (2D and range) is done using the strong weak learner that results form the learning step.

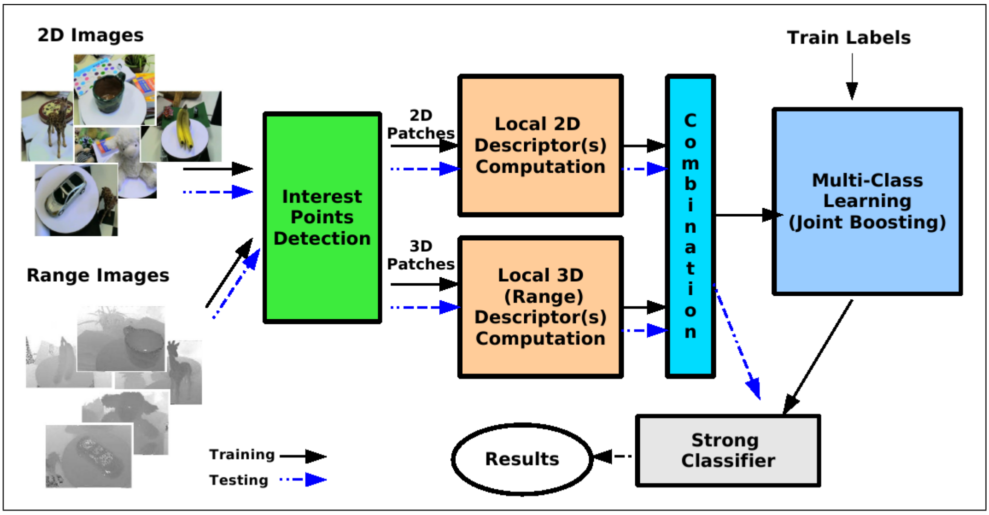

2D Images
Range Images
Interest Points Detection
2D Patches
3D Patches
Local 2D Descriptor(s) Computation
Local 3D (Range) Descriptor(s) Computation
Combination
Train Labels
Multi-Class Learning (Joint Boosting)
Strong Classifier
Results
Training
Testing

Figure 8.1: The general framework of the proposed GOR model based on different information fusion.

8.2.1 Local Description

Based on the type of the image (2D or range), a suitable point detector is used for local patches sampling. Then, suitable local descriptors are computed from the extracted regions. A set of different local descriptors is used for both image types including grayscale, color and shape descriptors.

2D Data

For the 2D images, an Hessian-Affine point detector [99, 100] is used as described in section 4.2.2. However, our implementation allows the use of any other point detector. A combination of two different types of local descriptors is then used: the SIFT descriptors [92] and the opponent color angle descriptors [148] (the same as described in section 4.2.3).

Range Data

A combination of the shape-specific local features presented in section 6.2.4 are used for describing range images. These descriptors are computed form local regions sampled in two different (separate) ways:

1. Using interest point detection as the procedure described in section 6.2.2.
2. Using the grid representation described in section 7.4.

However, our implementation allows the use of different local sampling and description methods.

8.2.2 Learning

The learning model described in section 7.2.2 for multi-class learning is used here. However, five different descriptors are combined instead of three as done in chapter 7. The advantage of our implementation of the learning model is that, it is flexible in the manner that it allows different number of descriptors to be combined. The same weak learner presented in section 7.2.2 is used here.

8.3 Performance Evaluations

The presented GOR model is evaluated experimentally to analyze its benefits and limitations. The performance is measured in three cases:

1. Using only appearance-based information for recognition.
2. Using only 3D range-based information.
3. Using a fusion of both different information types.

The model, for each previously mentioned case, is trained in three ways:

- Jointly with feature sharing among classes (JS).
- Jointly with no feature sharing among classes (JNS).
- Independently (Indp.) (see section 7.3.1).

8.3.1 Settings

Our dataset, JenaRange02, is used in all experimental evaluations. The settings of all experiments (*i.e.* number of iterations, training and test examples) is the same as mentioned in section 7.3.1.

8.3.2 Using Point Detection for Sampling Range Images

Recognition using range information only: see results in section 7.3.2, tables 7.1 and 7.2.

Recognition using appearance information only: The aim of this set of experiments is to measure the categorization performance of the model using only appearance-based information. A combination of the SIFT and color descriptors (SC) is used for learning and recognition. The recognition performance (classification rates at ROC-Equal-Error rates) using the test images is displayed in table 8.1 while table 8.3 presents the recognition confusion matrices. The use of appearance-based information yields better performance than using range-based information as shown in tables 8.1 and 7.1. However, it is still not able to solve the confusions between the different object classes as shown in table 8.3.

Recognition using a combination of appearance and range based information: To assess the performance of the model when different types of information (appearance and range) is used, a combination of the appearance and range based information (Shape-SIFT-color) is used for training and testing the recognition model. The recognition performance is shown in table 8.2, while table 8.4 displays the confusion matrices of recognition. The confusion using only range information is high in comparison to the case of using appearance information, while the confusion among the different object classes is notably reduced by using appearance-range (Shape-SIFT-color) combination. Again, the independent learning of the different classes reveals more robust performance than the other learning procedures.
Figure 8.2 displays comparisons of the recognition performance achieved by using appearance-based information only (SIFT-color combination), range-based information only (Shape) and appearance-range combination (Shape-SIFT-color) respectively. The figure displays the performance when the three different learning procedures are used. First, we can notice from the figure that each different type of information, either appearance-based (SIFT-color) or range-based (Shape) does not have robust performance over all different object classes. Range-based (Shape) information shows good performance over some classes such as class " Toys" while reveals bad performance over others such as " Fruits". For the appearance-range (SIFT-color) information, although it yields more robust performance than range-based (Shape), its performance varies from class to another. The use of appearance-range (Shape-SIFT-color) information combination improves the recognition performance over the use of range-based

Table 8.1: *Classification performance at the ROC-eqq.-err. rates of the multi-class recognition using SIFT-color features (SC) combination.* The model is trained jointly with feature sharing (Joint (Sharing)), jointly without feature sharing between the classes (Joint (No sharing)) and independently (Independent). The last column represent the average classification rates over the five classes.

Descriptors	Cars	Fruits	Animals	Toys	Cups	Avg. recog. over classes
Joint (Sharing)	73.30	75.00	78.00	65.00	68.30	71.90
Joint (No Sharing)	70.00	80.00	75.00	71.70	76.70	74.68
Independent	73.30	80.00	78.30	73.00	76.70	76.26

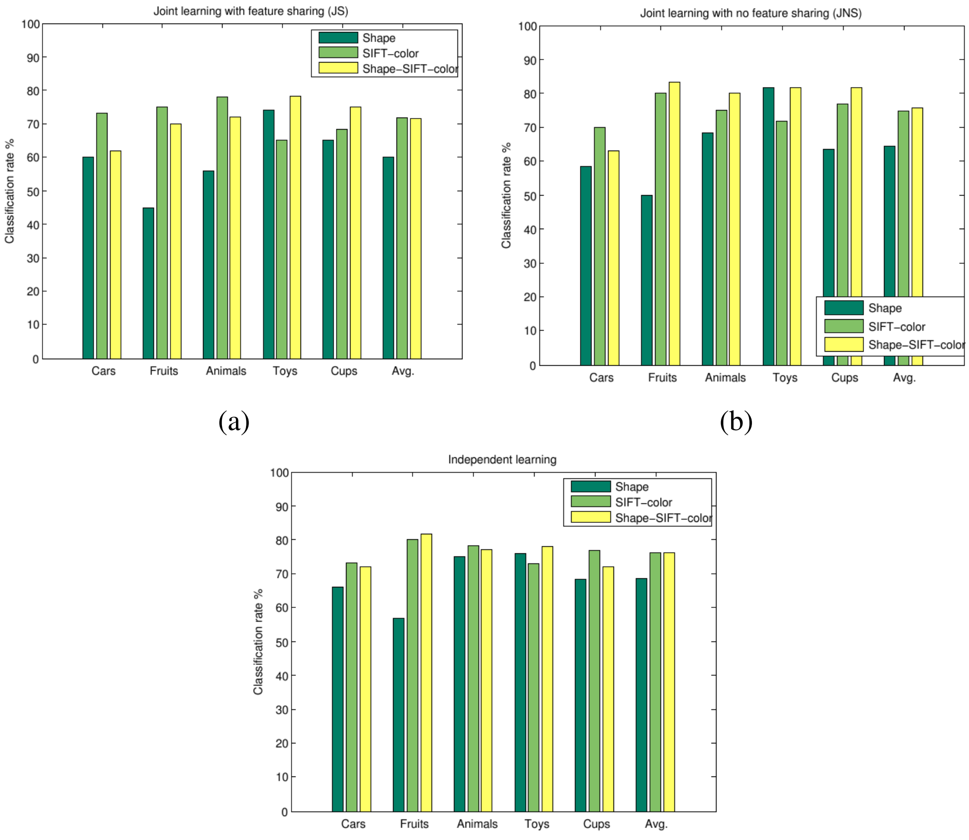
Table 8.2: *Classification performance at the ROC-eqq.-err. rates of the multi-class recognition using Shape-SIFT-color features combination.* The model is trained jointly with feature sharing (Joint (Sharing)), jointly without feature sharing between the classes (Joint (No sharing)) and independently (Independent). The last column represent the average classification rates over the five classes.

Descriptors	Cars	Fruits	Animals	Toys	Cups	Avg. recog. over classes
Joint (Sharing)	61.70	70.00	72.00	78.30	75.00	71.40
Joint (No Sharing)	63.00	83.30	680.000	81.70	81.70	75.60
Independent	72.00	81.70	77.00	78.00	72.00	76.14

(Shape) information for almost all the classes. However, when compared to the use of appearance-based (SIFT-color) information, improvements for some classes as well as for the overall recognition occur.

The results in Figure 8.2 reveal that the use of the different information combination in our model guarantees at least a robust recognition performance over all classes, if will not improve the recognition of some classes.

However, our fusion model suffers form the problem of *curse of dimensionality*, which affects the categorization performance of the information fusion and causes it to degrade when compared to the performance of each individual information type. More details of the curse of dimensionality problem will be discussed in the following section.



(a)

(b)

(c)

Figure 8.2: *Performance comparisons of categorization performance using range-based (Shape), appearance-based (SIFT-color) and appearance-range based (Shape-SIFT-color) combination respectively.* (a) Joint learning with feature sharing (JS). (b) Joint learning without feature sharing (JNS) and (c) Independent learning (Indp.). Local regions in range images are sampled using interest point detectors.

Table 8.3: *Confusion matrices of multi-class recognition using SIFT-color features (SC) combination*: Comparison of results using joint learning with feature sharing (Jointly (Sharing)), joint learning without feature sharing (Jointly (No Sharing)) and independent learning (Independently) respectively. For the computation of the confusion matrix, the best classification of an image over all classes is counted as the object category. Numbers represent percentage (%) of test images (60 images per class) classified for each class. Columns represent true classes.

	Jointly (Sharing)					Jointly (No Sharing)					Independently				
Class	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
Cars:c1	**62**	12	15	22	15	**47**	1	3	8	3	40	5	8	3	3
Fruits:c2	4	**72**	5	8	10	13	**77**	12	1	12	0	**52**	0	0	0
Animal:c3	18	2	**62**	6	28	15	2	33	1	5	15	0	**49**	0	5
Toys:c4	13	13	15	**52**	5	7	12	13	**53**	3	**45**	26	40	**94**	25
Cups:c5	3	1	3	12	**42**	18	8	**38**	27	**77**	0	17	3	3	**67**

Table 8.4: *Confusion matrices of multi-class recognition using Shape-SIFT-color features combination*: Comparison of results using joint learning with feature sharing (Jointly (Sharing)), joint learning without feature sharing (Jointly (No Sharing)) and independent learning (Independently) respectively. For the computation of the confusion matrix, the best classification of an image over all classes is counted as the object category. Numbers represent percentage (%) of test images (60 images per class) classified for each class. Columns represent true classes.

	Jointly (Sharing)					Jointly (No Sharing)					Independently				
Class	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
Cars:c1	**38**	5	3	0	0	**53**	5	5	5	7	**65**	10	8	7	8
Fruits:c2	2	**64**	3	3	0	15	**80**	8	5	3	0	**72**	0	0	0
Animal:c3	12	0	32	5	12	3	0	**32**	0	7	12	3	**59**	1	2
Toys:c4	23	23	25	**89**	21	12	5	23	**78**	5	20	15	30	**92**	20
Cups:c5	25	8	**37**	3	**67**	17	10	**32**	12	**78**	3	0	3	0	**70**

8.3.3 Using Regular Grid for Sampling Range Images

Now, the experiments using the appearance-range based combination is repeated with applying the grid sampling to range data instead of interest point detection according the procedure described in section 7.4. Grid sampling with patch size $N = 20 \times 20$ and with patches overlap ($M = \frac{N}{2}$) is used. Note that the grid sampling is only applied to the used range images and is not applied to the 2D images. The experiments are performed with the same settings mentioned in section 8.3.1.

Recognition using range information only: The categorization results using the range-based (Shape) information with grid sampling of range images were presented in section 7.4 in tables 7.4 and 7.3.

Recognition using appearance information only: The recognition results of the appearance-range based (SIFT-color) information combination are those presented in tables 8.1 and 8.3.

Recognition using a combination of appearance and range based information: The grid sampling of range images improved the performance of range-based (Shape) information and, hence, improved the recognition performance of the appearance-range information combination. The use of appearance-range combination revealed high performance gain for some classed as for the average recognition over the classes as shown in table 8.5. The confusion among the different classes is further reduced using the combination. This confusion reduction is not only for learning the different classes independently, but also with learning using the other learning procedures as well. This can also be noticed from Figure 8.4. We can notice the performance gain using the range-based information with grid sampling of range images. The performance using the range-based information exceeds the performance using appearance-based for some classes such as classes ”toys ” and ”Fruits” (with independent learning). However, the use of appearance-range based information fusion still reveals robust performance for all the classes.

Figure 8.3 displays examples of incorrectly classified images of each object class. The images are simple to be miss-classified while the model is able to correctly classify more complex images. However, still some plausible reasons exist for the classification failure of some images such as the existence of an instance of the predicted class in the images.

Curse of Dimensionality Phenomena [69] The performance of a classifier depends of the interrelationship between three factors: sample size, number of features and classifier complexity. It is well known that the probability of missclassification does not increase as the number of features increases. However, it has often been observed in practice that, the added features may lead to actual performance degradation of the classifier if the number of used training samples is small relative to the feature size. All commonly used classifiers can suffer from the curse of dimensionality. However, some guidelines have been suggested regarding the ration of the sample size and the number of features to avoid the curse of dimensionality. Using at least ten times as many training examples per class as the number of features in designing a classifier can avoid this problem. However, the more complex the classifier is, the larger should the ration of sample size and feature size be.

Therefore, the performance degradation occurred when more features are combined in our classifier is due the curse of dimensionality problem. Although the sample size to features number ration per class in is higher than ten in our model, still our classifier suffers from the problem, since it is not a simple one. One solution to avoid this problem is to increase the number of training samples per class. However, further investigations should be done to estimate the appropriate sample size for our problem which will help to overcome this problem.

Table 8.5: *Classification performance at the ROC-eqq.-err. rates of the multi-class recognition using Shape-SIFT-color features combination.* Grid representation of range images is used. The dense grid patches are of size $N = 20 \times 20$ with patches overlap ($M = \frac{N}{2}$). The model is either trained jointly with feature sharing (Joint (Sharing)), jointly without feature sharing between the classes (Joint (No sharing)) or independently (Independent). The last column presents the average classification rates over the five classes.

Descriptors	Cars	Fruits	Animals	Toys	Cups	Avg. recog. over classes
Joint (Sharing)	76.00	82.00	72.00	85.00	75.00	78.00
Joint (No Sharing)	77.00	81.67	73.00	91.67	72.00	79.13
Independent	86.70	86.70	75.00	90.00	82.00	84.08

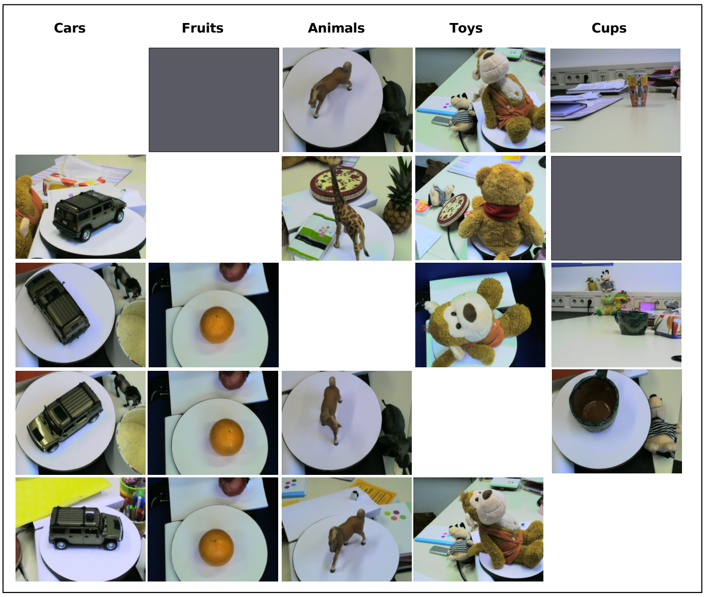
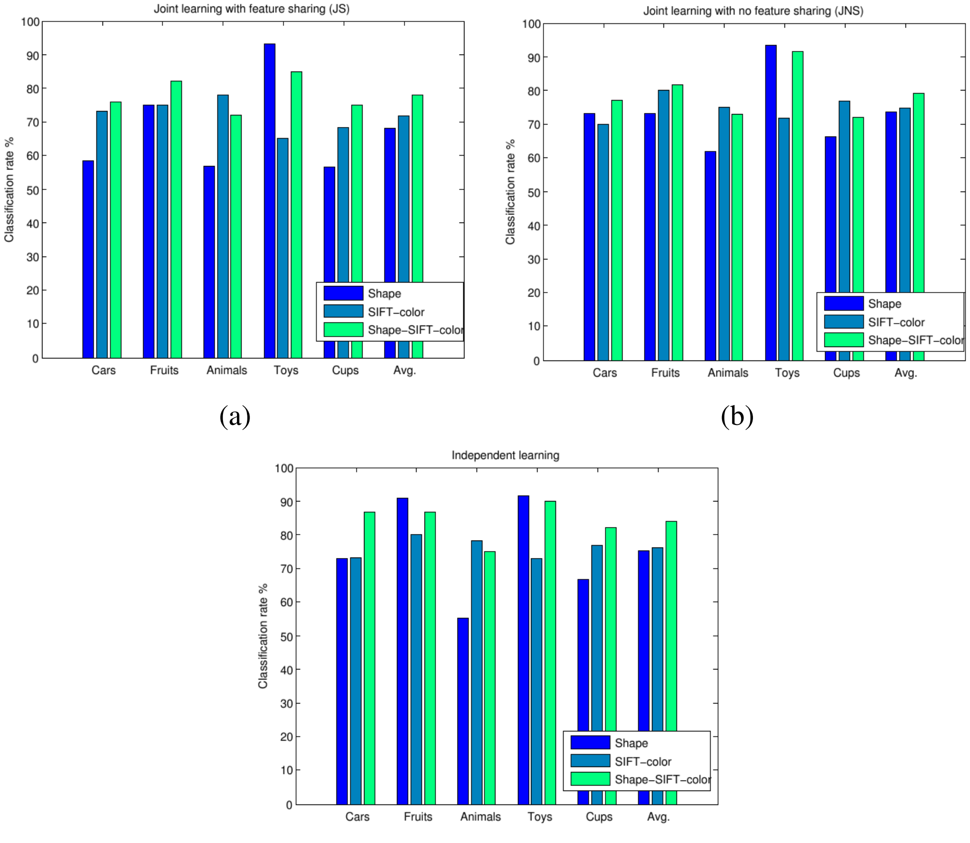


Figure 8.3: *Examples of incorrectly classified images from JenaRange02 dataset using appearance and range-based information fusion.* Columns represent true (actual) classes while rows represent predicted classes: cars (first row), fruits (second row), animals (third row), toys (fourth row) and cups (last row).



(a) (b)

(c)

Figure 8.4: *Performance comparisons of categorization performance using range-based (Shape), appearance-based (SIFT-color) and appearance-range based (Shape-SIFT-color) combination respectively.* (a) Joint learning with feature sharing (JS). (b) Joint learning without feature sharing (JNS) and (c) Independent learning (Indp.). Local regions in range images are sampled using dense regular grid.

Table 8.6: *Confusion matrices of multi-class recognition using Shape-SIFT-color features combination*: Comparison of results with joint learning with feature sharing (Jointly (Sharing)), joint learning without feature sharing (Jointly (No Sharing)) and independent learning (Independently) respectively. Grid representation of range images is used with patches size $N = 20 \times 20$ and patches overlap ($M = \frac{N}{2}$). For the computation of the confusion matrix, the best classification of an image over all classes is counted as the object category. Numbers represent percentage (%) of test images (60 images per class) classified for each class. Columns represent true classes.

	Jointly (Sharing)					Jointly (No Sharing)					Independently				
Class	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
Cars:c1	**54**	2	10	0	7	**57**	8	13	5	15	**73**	0	7	3	7
Fruits:c2	13	**82**	13	3	0	7	**77**	2	0	0	2	**87**	3	2	0
Animal:c3	22	2	**49**	0	15	26	3	**55**	2	18	20	2	**67**	2	6
Toys:c4	8	11	25	**94**	21	3	5	12	**90**	10	2	8	13	**90**	12
Cups:c5	3	3	3	3	**57**	7	7	18	3	**57**	3	3	10	3	**75**

8.4 Conclusions

In this chapter we have presented a first step toward fusing different information cues extracted from different image types for generic 3D object recognition. We have combined our appearance-based model for recognition from 2D images presented in chapter 4 and our range-based model presented in chapters 6 and 7 in one model for multi-class recognition of generic classes of 3D objects. Robust categorization performance has been revealed by the new model. The experiments have shown how range-based information benefits from the additional information given by the appearance based and vis versa. The experiments have also shown how robust the categorization is for all the object classes when the combination is used. However, this combination increases also the computation time.

This is a quite rough combination among the different information cues and there is a room for improvements is possible.

Chapter 9

Summary, Conclusions and Future work

This chapter summarizes the work presented throughout this thesis with resuming its main contributions and outcomes. Then, conclusions on the obtained results are drawn. Finally, a discussion to possible improvements and future work is given.

9.1 Work Summary

In this thesis, we have addressed different GOR problems (2D and 3D GOR) using different information cues from still images. We have presented three main models.

First, a model based on a combination of local appearance information for generic recognition of 2D images is introduced. The model exploits a combination of local texture and color information for binary classification of generic object classes. The framework of the model learns a classifier for each object class in a weakly supervised manner. The performance of the model has been investigated using two famous GOR benchmarks and robust recognition results have been obtained.

The second model addresses the more difficult problem of generic 3D object recognition. The model is based on local description of range images using simple shape features for recognition. First, interest regions are detected and extracted from range images and then are described using a combination of three simple local shape descriptors. Afterwards, learning is achieved using a boosting algorithm. The general framework of the model using range images is new and never been applied before. However, promising performance has been revealed by the model.

The third model is one for multi-class recognition of generic 3D objects from range images. It is an extension to our generic 3D recognition model which tackles a more difficult task. Range images are still used for recognition. The Joint Boosting algorithm is modified to cope with our different image type (range images) and weak learner.

Finally, we have presented a model which makes use of different information cues for multi-class recognition of generic 3D objects. The models exploits a combination of appearance-based information extracted from 2D images and shape-based information extracted from range images for recognition. Actually, the model combines our model for 2D object recognition together with our model for generic 3D object recognition in one model suitable for both tasks.

Moreover, to be able to address the problem of generic 3D object recognition from range images, we have constructed a new object category dataset which provides different data (image) types (2D color and range images) about its member classes. The dataset is considered to be the first one to provide range images for different object classes. Moreover, it contains images of complex scenes, which makes it a challenging dataset.

9.2 Contributions and Outcomes

This thesis has presented different contributions:

1. A generic 2D object recognition model based on appearance information by combining different appearance cues, namely texture and color, for recognition.
2. A novel GOR dataset which provides different image types, 2D (color) and range images, of complex nature of different object classes.
3. A novel model for generic 3D object recognition from range images and its extension to the more difficult task of multi-class recognition.
4. A novel model for multi-class generic 3D object recognition using a combination of different information cues (appearance and shape-based information) extracted from different image types (2D and range images).

However, there are specific outcomes have been achieved in this thesis:

- We achieved very good and robust recognition performance of 2D objects on the most complex image set (at this time) using only local appearance information with weak supervision.

- Investigated the role of the counter-class used for the binary classification task for affecting the final recognition performance of the model. The use of counter-classes with simple images, that have different context from the images of the object classes, do not reveal the actual recognition performance of the model.

- Compared the performance of two famous texture descriptors in the context of generic 2D object recognition.

- Evaluated the performance of the new Boosting algorithm, SoftBoost algorithm, in real world application (the generic 2D object recognition problem) for the first time [1] using noise free as well as noisy data. Moreover, we assessed its performance in solving the problem by establishing comparisons to the famous AdaBoost algorithm.

- We have presented a novel GOR dataset which provide 2D and 3D (range) images bout different object classes. Moreover, a performance evaluation of the dataset and comparisons to a famous GOR benchmark reveals the challenging nature of our new dataset.

- Used the regular grid sampling method for sampling local patches from range images, which is a new sampling method for range images. This sampling method has revealed good performance for recognition using range-based information.

9.2.1 Summarization of Best Results

This section aims to provide a summarization of the best classification and recognition results achieved throughout this thesis. The best results achieved by each recognition model mentioned in the thesis will be summarized on the basis of:

- Actually recognition rates (see section 2.5 for definition).

- Time needed for training and testing each recognition model. Training time is divided into: time to compute distances between images and time needed for training by the boosting algorithm. The training and test time represent the time needed for a whole set of training and test example, not for an individual image.

---

[1] We were the first to evaluate the performance of SoftBoost algorithm in the time we carried out our experiments.

Generic 2D Object Recognition Model The performance of the generic 2D object recognition model presented in chapter 4 has been evaluated on two datasets: Caltech 4 and Graz02. On the Caltech dataset, the SIFT-color combination (SC-G) has achieved the best classification rates which are summarized in table 9.1 together with the required training and test time.

Table 9.1: Best results achieved on Caltech 4 dataset.

Class	Recognition rate	No. of test images
Motor	96.00 %	100
Cars	100.00 %	100
Airplanes	88.00 %	100
Faces	96.00%	100
Time required for distances computation for each class	1.84 hours	
Training time for each class	100 seconds (1 sec. /iteration)	
Test time for each class	0.9 minute	

For the Graz02 dataset, the best results have been achieved using the SIFT-color combination (SC). Table 9.2 summarizes the best results as well as the need compation time.

Table 9.2: Best results achieved on Graz02 dataset.

Class	Recognition rate	No. of test images
Bikes	82.00 %	150
Cars	77.33 %	150
Persons	82.67 %	150
Time required for distances computation for each class	3.35 hours	
Training time for each class	300 seconds (2 sec. /iteration)	
Test time for each class	1.2 minute	

Generic 3D Object Recognition Model Our dataset JenaRange01 has been used to evaluate the performance of the model for generic 3D object recognition from range images mentioned in chapter 6. The best results have been achieved in the categorization experiments (experiment 1). Table 9.3 summarizes the recognition rates achieved for each object class, in addition to the required training and test time.

Table 9.3: Best results achieved on JenaRange01 dataset dataset.

Class	Recognition rate	No. of test images
Cars	96.00 %	100
Motors	98.00 %	100
Animals	100.00 %	100
Time required for distances computation for each class	30 minutes	
Training time for each class	26 minutes (15.6 sec. /iteration)	
Test time for each class	0.7 seconds	

Multi-Class Generic 3D Object Recognition Model For the multi-class recognition from range images model presented in chapter 7, all performance evaluation experiments are accomplished using our JaneRange02 dataset. The best categorization results have been achieved with using dense grid as a sampling method for range images (with independent learning). The recognition rates as well as the needed computation time are summarized in table 9.4. It should be noted that the required computation time is for the whole model.

Fusion model for Multi-Class Generic 3D Object Recognition JenaRange02 dataset is also used in all evaluations performed to investigate the performance of the fusion model presented in chapter 8. The best results have been achieved using the appearance and range-based information fusion with grid sampling of range images (with independent learning). The achieved recognition rates and the needed computation time are summarized in table 9.5.

Table 9.4: Best results achieved on JenaRange02 dataset dataset for multi-class recognition using range-based information.

Class	Recognition rate	No. of test images
Cars	48.00%	60
Fruits	92.00 %	60
Animals	59.00 %	60
Toys	70.00 %	60
Cups	45.00 %	60
Total	62.80 %	300
Time required for distances computation	24 hours	
Training time	3.6750 hours (88.2 sec./iteration)	
Test time	8 minutes	

Table 9.5: Best results achieved on JenaRange02 dataset dataset for multi-class recognition using appearance and range-based information fusion.

Class	Recognition rate	No. of test images
Cars	73.00%	60
Fruits	87.00 %	60
Animals	67.00 %	60
Toys	90.00 %	60
Cups	75.00 %	60
Total	78.40 %	300
Time required for distances computation	24 hours	
Training time	6.1250 hours (147 sec./iteration)	
Test time	15 minutes	

9.3 Conclusions

It is not a long way for any artificial GOR system to be comparable to the human visual system. The work presented in this thesis show different possibilities where the task of GOR can be successfully solved. However, this is still unfortunately with some restrictions.

The previous section has listed the contributions and outcomes achieved throughout this thesis. However, to be realistic, we should also shed light on the limitations and drawbacks exist in the presented methods.

Our model for generic recognition of 2D objects from 2D images revealed very robust performance in classification while it does not perform any detection or localization. The missing ability to localize objects in images is a limitation of our model. Moreover, the learning step in the model is slow which, affects the real time performance of the model.

The generic 3D object recognition model from range images is a novel model that address this difficult problem from range images. Although the general framework of the model is simple, good recognition performance is achieved. However, the used shape descriptors are very simple and do not comprise enough information about the shape of the different object classes. More robust shape description method should be used. Moreover, more robust local sampling methods of range images should also be investigated in order to improve the performance. These two problems are more clear when the recognition model is extended to the multi-class recognition case, where differentiating among different shape classes is required. However, two aspects should be put into considerations when assessing the model's performance: the difficult nature of the original problem and the noisy nature of the used range images.

The multi-class generic 3D object recognition model has another additional limitation that is learning is getting more slower and complex when the number of classes increases. As we live in a world full of thousands of different object categories, learning these categories in a simple and fast way is a mandatory requirement for a robust real time GOR system.

The fusion of appearance and range-based information for multi-class GOR has been investigated where promising results have been obtained. There are much room for improvements to be done here such as investigating different ways of combining the different information cues for achieving better performance.

Finally, these thesis has presented several possible steps towards solving the difficult problem of generic 2D object recognition. Moreover, it has presented first steps in addressing and, hence solving the more difficult problem of generic 3D object recognition from range images. However, the limitations exist in the different models presented in this thesis could be used as steps for more improvements and, hence providing more robust solutions to the problem.

9.4 Future work

There are several ways where the research could go further. Following, we discuss them briefly [2]:

Extending the 2D recognition model : To extend the model to be able to perform localization as well by adding some spatial information among the used descriptors.

Improving the 3D recognition model : One field for possible future work is continue addressing the generic 3D object recognition from range images by finding and investigating suitable method to locally describe the range images. Moreover, adding the possibility of pose estimation to the 3D model is an important step. Finding a more proper and robust way to combine different information cues is another important modification suggestion. Also, enhancing the TOF images for the task of robust recognition is another important task to be accomplished.

Online Learning Adding the ability to the recognition model to learn new object categories online would be a good and valuable extension to our 3D recognition model.

Real-time Model We could also improve the performance of the model to make it suitable for robust real time recognition. Although we have already built a real-time model, it is still a slow and not totally robust one.

---

[2] Actually, there is a wide room for future work in the area of GOR. However, we present her some of those related to our models.

Bibliography

[1] http://www.mis.informatik.tu-darmstadt.de/Research/Projects/categorization/eth80-db.html.

[2] http://www.vision.caltech.edu/html-files/archive.html.

[3] http://l2r.cs.uiuc.edu/˜cogcomp/Data/Car/.

[4] http://www.emt.tugraz.at/˜pinz/data/GRAZ$_$01/.

[5] http://www.emt.tugraz.at/˜pinz/data/GRAZ$_$02/.

[6] http://vangogh.ai.uiuc.edu/silvio/3Ddataset.html.

[7] http://sampl.ece.ohio-state.edu/data/3DDB/RID/index.htm.

[8] http://range.informatik.uni-stuttgart.de/.

[9] http://www.vision.caltech.edu/Image_Datasets/Caltech101/.

[10] M. S. Abd-Al-Wahaab, S. F. Bahgat, A. S. Hussein, and D. A. Hegazy. Three-dimensional object recognition using a combination of hu moment and affine moment invariants features. In *Proceedings of IEEE Mediterranean Microwave Symposium (MMS'03)*, pages 279–287, 2003.

[11] M. S. Abd-Al-Wahaab, S. F. Bahgat, A. S. Hussein, and D. A. Hegazy. View-based 3-d object recognition using color features. In *Proceedings of International Arab Conference on Information Technology*, volume 2, pages 747–752, 2003.

[12] Mohammed S. Abdel-Wahaab, Sayed F. Bahgat, Ashraf S. Hussein, and Doaa M. Hegazy. Three-dimensional object recognition using support vector machine neural network based on moment invariant features. In *Proceedings of 5th International Conference on Enterprise Information Systems (ICEIS03)*, volume 2, pages 583–588, 2003.

[13] Shivani Agarwal and Dan Roth. Learning a Sparse Representation for Object Detection. In *7th European Conference on Computer Vision ECCV02*, volume 2353, pages 113–130, London, UK, May 2002. Springer-Verlag.

[14] G. J. Agin. Computer vision systems for industrial inspection and assembly. *Computer*, 13(5):11–20, 1980.

[15] Jaume Amores, Nicu Sebe, and Petia Radeva. Fast spatial pattern discovery integrating boosting with constellations of contextual descriptors. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 769–774, Washington, DC, USA, 2005. IEEE Computer Society.

[16] Juan Andrade-cetto and Avinash C. Kak. Object recognition. *Wiley Encyclopedia of Electrical and Electronics Engineering*, 1:2000, 2000.

[17] A. P. Ashbrook, N. A. Thacker, P. I. Rockett, and C. I. Brown. Robust recognition of scaled shapes using pairwise geometric histograms. In *BMVC '95: Proceedings of the 6th British conference on Machine vision (Vol. 2)*, pages 503–512. BMVA Press, 1995.

[18] D. H. Ballard. *Generalizing the hough transform to detect arbitrary shapes*, pages 714–725. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

[19] Aharon Bar-Hillel, Tomer Hertz, and Daphna Weinshall. Object class recognition by boosting a part-based model. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 702–709, Washington, DC, USA, 2005. IEEE Computer Society.

[20] Ronen Basri. Recognition by prototypes. *International Journal of Computer Vision*, 19:19–147, 1992.

[21] P.R. Beaudet. Rotationally invariant image operators. In *Proceedings of International Conference on Pattern Recognition*, pages 579–583, 1978.

[22] Serge Belongie, Jitendra Malik, and Jan Puzicha. Matching shapes. In *International Conference on Computer Vision*, pages 454–461, 2001.

[23] Serge Belongie, Jitendra malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:509–522, 2002.

[24] Alexander Berg, Tamara Berg, and Jitendra Malik. Shape matching and object recognition using low distortion correspondence. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[25] Paul J. Besl and Ramesh C. Jain. Three-dimensional object recognition. *ACM Comput. Surv.*, 17(1):75–145, 1985.

[26] Anna Bosch, Andrew Zisserman, and Xavier Muoz. Scene classification using a hybrid generative/discriminative approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(4):712–727, 2008.

[27] Rodney Allen Brooks. Symbolic reasoning among 3-d models and 2-d images. *Artificial Intelligences*, 17(1-3):285–348, August 1981.

[28] B. Büttgen, T. Oggier, M. Lehmann, R. Kaufmann, and F. Lustenberger. Ccd/cmos lock-in pixel for range imaging: Challenges, limitations and state-of-the-art. In *1st Range Imaging Research Days, ETH,Zurich, Switzerland*, pages 21–32, 2005.

[29] Barbara Caputo, Christian Wallraven, and Maria-Elena Nilsback. Object categorization via local kernels. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 2*, pages 132–135, Washington, DC, USA, 2004. IEEE Computer Society.

[30] H. Chen and B. Bhanu. 3d free-form object recognition in range images using local surface patches. *Pattern Recognition Letters*, 28(10):1252–1262, July 2007.

[31] Jin-Long Chen and George C. Stockman. Indexing to 3d model aspects using 2d contour features. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:913, 1996.

[32] Chin S. Chua and Ray Jarvis. Point signatures: A new representation for 3d object recognition. *International Journal of Computer Vision*, 25(1):63–85, October 1997.

[33] James L. Crowley and Alice C. Parker. A representation for shape based on peaks and ridges in the difference of low-pass transform. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(2):156–170, March 1984.

[34] Gabriella Csurka, Christopher Bray, Cedric Dance, and Lixin Fan. Visual categorization with bags of keypoints. In *European Conference of Computer Vision Workshop on Stat. Learning in Computer Vision*, pages 59–74, 2004.

[35] Ayhan Demiriz, Kristin P. Bennett, and John Shawe-Taylor. Linear programming boosting via column generation. *Machine Learning*, 46(1-3):225–254, 2002.

[36] Chitra Dorai and Anil K. Jain. COSMOS - a representation scheme for 3d free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 190(10):1115–1130, 1997.

[37] Gyuri Dorkó and Cordelia Schmid. Object Class Recognition Using Discriminative Local Features. Technical Report RR-5497, INRIA - Rhône-Alpes, February 2005.

[38] Richard O. Duda, Peter E. Hart, and David C. Strok. *Pattern Classification*. Wiley, 2001.

[39] W. Eric, W. Eric L. Grimson, and Daniel P. Huttenlocher. On the sensitivity of the hough transform for object recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12, 1990.

[40] Tom Fawcett. Roc graphs: Notes and practical considerations for researchers, 2004.

[41] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Pattern Recognition Workshop*, page 178, 2004.

[42] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61:55–79, 2005.

[43] Rob Fergus, Pietro Perona, and Andrew Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Computer Society Conference on computer vision and Pattern Recognition CVPR3*, volume 2, pages 264–271, June 2003.

[44] Rob Fergus, Pietro Perona, and Andrew Zisserman. A visual category filter for google images. *Proc. ECCV*, 2004.

[45] Rob Fergus, Pietro Perona, and Andrew Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 380–387, Washington, DC, USA, 2005. IEEE Computer Society.

[46] Vittorie Ferrari, Tinne Tuytelaars, and Luc Van Gool. Simultaenous object recognition and segmentation by image exploration. In *8th European Conference on Computer Vision ECCV*, 2004.

[47] J. Flusser and T. Suk. Pattern recognition by affine moment invariants. *Pattern Recognition*, 26(1):167–174, January 1993.

[48] Wolfgang Forstner. A feature based correspondence algorithm for image matching. In *IEEE International Symposium on Signal Processing and Information Technology*, pages III: 150–16, 1986.

[49] William T. Freeman and Edward H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:891–906, 1991.

[50] Yoav. Freund and Robert E. Schapire. A decision theoretic generalization of online learning and application to boosting. *Computer and System Sciences*, 55(1):119–139, 1997.

[51] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28:2000, 1998.

[52] Mario Fritz, Bastian Leibe, Barbara Caputo, and Bernt Schiele. Integrating representative and discriminant models for object category detection. In *In ICCV*, pages 1363–1370, 2005.

[53] Andrea Frome, Daniel Huber, Ravi Kolluri, Thomas Bulow, and Jitendra Malik. Recognizing objects in range data using regional point descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, May 2004.

[54] Seyed Eghbal Ghobadi, Omar Edmond Loepprich, Oliver Lottner, Farid Ahmadov, Klaus Hartmann, Wolfgang Weihs, and Otmar Loffeld. Analysis of the personnel safety in a man-machine-cooperation using 2d/3d images. In *Proceedings of the EURON/IARP International Workshop on Robotics for Risky Interventions and Surveillance of the Environment*, Benicassim, Spain, January 2008.

[55] Kristen Grauman and Trevor Darrell. The pyramid match kernel: Efficient learning with sets of features. *Journal of Machine Learning*, 8:725–760, 2007.

[56] Cris Harris and Mike Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[57] Parag Havaldar, Gerard Medioni, and F.ridtjof Stein. Perceptual grouping for generic recognition. *International Journal of Computer Vision*, 20(1/2):59–80, 1996.

[58] Doaa Hegazy and Joachim Denzler. Boosting local colored features for generic object recognition. *Pattern Recognition and Image Understanding*, 18:323–327, 2008.

[59] Doaa Hegazy and Joachim Denzler. *Generic Object Recognition using Boosted Combined Features*, volume 4931 of *Lecture Notes in Computer Science/ Robot Vision*, chapter Robot Vision 27, pages 355–366. Springer Berlin / Heidelberg, 2008.

[60] Doaa Hegazy and Joachim Denzler. Performance comparison and evaluation of adaboost and softboost algorithms on generic object recognition. In *Proceedings of 5th International Conference on Machine Learning and Pattern Recognition (MLPR08)*, volume 35, pages 70–74, 2008.

[61] Doaa Hegazy and Joachim Denzler. Combining appearance and range based information for multi-class generic object recognition. In *Proceedings of 14th Iberoamerican Congress on Pattern Recognition (CIARP09)- To appear*, 2009.

[62] Doaa Hegazy and Joachim Denzler. Generic 3d object recognition from time-of-flight images using boosted combined shape features. In *Proceedings of International Conference on Computer Vision, Theory and Applications (VISAPP 09)*, 2009.

[63] Guenter Hetzel, Bastian Leibe, Paul Levi, and Bernt Schiele. 3d object recognition from range images using local feature histograms. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'01)*, volume 2, pages 394–399, 2001.

[64] Bar A. Hillel, T. Hertz, and D. Weinshall. Object class recognition by boosting a part-based model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 702–709, 2005.

[65] Ming K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, IT-8:179–187, February 1962.

[66] Stephan Hussmann, Thorsten Ringbeck, and Bianca Hagebeuker. *A Performance Review of 3D TOF Vision Systems in Comparison to Stereo Vision Systems*. I-Tech, 2008.

[67] Daniel P. Huttenlocher and Liana M. Lorigo. Recognizing three-dimensional objects by comparing two-dimensional images. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:878, 1996.

[68] Daniel P. Huttenlocher and Shimon Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, November 1990.

[69] Anil K. Jain, Robert P. W. Duin, and Jianchang Moa. (statistical pattern recognition: A review). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 2000.

[70] Andrew Johnson and Martial Hebert. *Using spin images for efficient object recognition in cluttered 3D scenes*, 21(1):433 – 449, May 1999.

[71] Andrew E. Johnson and Martial Hebert. Recognizing objects by matching oriented points. In *In CVPR*, pages 684–689, 1996.

[72] Frederic Jurie and Bill Triggs. Creating efficient codebooks for visual recognition. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 604–610, Washington, DC, USA, 2005. IEEE Computer Society.

[73] Timor Kadir and Michael Brady. Saliency, scale and image description. *International Journal of Computer Vision*, V45(2):83–105, November 2001.

[74] Timor Kadir, Andrew Zisserman, and Michael Brady. An affine invariant salient region detector. In *European Conference of Computer Vision*, 2004.

[75] Jan J. Koenderink and Andrea J. van Doorn. Representation of local geometry in the visual system. *Biol. Cybern.*, 55(6):367–375, 1987.

[76] Jan J. Koenderink and Andrea J. van Doorn. Surface shape and curvature scales. *Image and Vision Computing*, 10(8):557–564, October 1992.

[77] Pawan M. Kumar, Philip Torr, and Andrew Zisserman. Extending pictorial structures for object recognition. In *British Machine Vision Conference*, 2004.

[78] A. Kushal, C. Schmid, and J. Ponce. Flexible object models for category-level 3d object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[79] Robert Lange. *3D Time-of-Flight Distance Measurement with Custom Solid-State Image Sensors in CMOS/CCD-Technology*. 2000. PhD thesis, University of Siegen.

[80] Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *In ECCV workshop on statistical learning in computer vision*, pages 17–32, 2004.

[81] Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *In ECCV workshop on statistical learning in computer vision*, pages 17–32, 2004.

[82] Bastian Leibe and Bernt Schiele. Analyzing appearance and contour based methods for object categorization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[83] Bastian Leibe and Bernt Schiele. Scale-invariant object categorization using a scale-adaptive mean-shift search. pages 145–153, 2004.

[84] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 878–885, Washington, DC, USA, 2005. IEEE Computer Society.

[85] Aleš Leonardis and Horst Bischof. Robust recognition using eigenimages. *Computer Vision and Image Understanding*, 78(1):99–118, 2000.

[86] Thomas K. Leung, Michael C. Burl, and Pietro Perona. Probabilistic affine invariants for recognition. In *In Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn*, pages 678–684, 1998.

[87] Xinju Li and Igor Guskov. Multi-scale features for approximate alignment of point-based surfaces. In *SGP '05: Proceedings of the third Eurographics symposium on Geometry processing*, page 217, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.

[88] Xinju Li and Igor Guskov. 3d object recognition from range images using pyramid matching. In *International Conference on Computer Vision*, pages 1–6, 2007.

[89] Sven Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31:983–1001, 1998.

[90] David G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

[91] David G. Lowe. Object Recognition from Local Scale-Invariant Features. In *International Conference on Computer Vision*, pages 1150–1157, September 1999.

[92] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[93] Raphaël Marée, Pierre Geurts, Justus Piater, and Louis Wehenkel. Random subwindows for robust image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 34–40, June 2005.

[94] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *In British Machine Vision Conference*, volume 1, pages 384–393, 2002.

[95] Jiri Matas, Dimitri Koubaroulis, and Josef Kittler. Colour image retrieval and object recognition using the multimodal neighbourhood signature. In *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part I*, pages 48–64, London, UK, 2000. Springer-Verlag.

[96] Ajmal S. Mian, Mohammed Bennamoun, and Robyn Owens. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1584–1601, 2006.

[97] Ajmal S. Mian, Mohammed Bennamoun, and Robyn A. Owens. Automatic correspondence for 3d modeling: an extensive review. *International Journal of Shape Modeling*, 11(2):253–291, 2005.

[98] Krystian Mikolajczyk, Bastian Leibe, and Bernt Schiele. Local features for object class recognition. *Computer Vision, IEEE International Conference on*, 2:1792–1799, 2005.

[99] Krystian Mikolajczyk and Cordelia Schmid. An affine invariant interest point detector. In *7th European Conference on Computer Vision ECCV02*, pages 128–142, 2002.

[100] Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 1(60):63–86, 2004.

[101] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI*, 27(10):1615–1630, 2005.

[102] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frediek schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 2005.

[103] Florica Mindru, Tinne Tuytelaars, Luc Van Gool, and Theo Moons. Moment invariants for recognition under changing viewpoint and illumination. *Computer Vision and Image Understanding*, 94:1–30, 2004.

[104] Frank Moosmann, Diane Larlus, and Frédéric Jurie. Learning saliency maps for object categorization. In *ECCV International Workshop on The Representation and Use of Prior Knowledge in Vision*. Springer, 2006.

[105] Frank Moosmann, Eric Nowak, and Frédéric Jurie. Randomized clustering forests for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1632–1646, 2008.

[106] Pierre Moreels and Pietro Perona. Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision*, 73(3):263–284, 2007.

[107] H. Murase and S.K. Nayar. Visual learning and recognition of 3d object from appearance. *International Journal of Computer Vision*, 14(1):5–24, January 1995.

[108] Jim Mutch and David G. Lowe. Object class recognition and localization using sparse features with limited receptive fields. *International Journal of Computer Vision*, 80(1):45–57, 2008.

[109] D. Van Nieuwenhove, W. Van der Tempel, r. Grootjans, and M. kuijk. Time-of-flight optical ranging sensor based on a current assisted photonic demodulator. In *Symposium IEEE/LEOS Benelux Chapter*, pages 209–212, 2006.

[110] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[111] Andreas Opelt. *Generic Object Recognition*. 2006. PhD thesis, University of Graz.

[112] Andreas Opelt, Michael Fussenegger, Axel Pinz, and Peter Auer. Weak hypotheses and boosting for generic object detection and recognition. In *European Conference of Computer Vision*, 2004.

[113] Andreas Opelt and Axel Pinz. Object localization with boosting and weak supervision for generic object recognition. In *Scandinavian Conference on Image Analysis SCIA05*, pages 862–871, 2005.

[114] Andreas Opelt, Axel Pinz, Michael Fussenegger, and Peter Auer. Generic object recognirion with boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):416–431, 2006.

[115] Andreas Opelt, Axel Pinz, and Andrew Zisserman. Incremental learning of object detectors using a visual shape alphabet. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3–10, Washington, DC, USA, 2006. IEEE Computer Society.

[116] Andreas Opelt and Andrew Zisserman. A boundary-fragment-model for object detection. In *European Conference of Computer Vision*, pages 575–588, 2006.

[117] Stephen Palmer, Eleanor Rosch, and Paul Chase. *Canonical perspective and the perception of objects*. Erlbaum Hillsdale.

[118] Axel Pinz. Object categorization. *Foundations and Trends in Computer Graphics and Vision*, 1(4):255–353, 2006.

[119] Jean Ponce, T. L. Berg, M. Everingham, D. Forsyth, M. Hebert, Svetlana Lazebnik, Marcin Marszałek, Cordelia Schmid, C. Russell, A. Torralba, C. Williams, Jianguo Zhang, and Andrew Zisserman. Dataset issues in object recognition. In *Towards Category-Level Object Recognition*, pages 29–48. Springer, 2006.

[120] Jean Ponce, Tamara L. Berg, Mark Everingham, David A. Forsyth, Martial Hebert, Svetlana Lazebnik, Marcin Marszalek, Cordelia Schmid, Bryan C. Russell, A. Torralba, Christopher K. I. Williams, Jianguo Zhang, and Andrew Zisserman. Dataset issues in object recognition. In Jean Ponce, Martial Hebert, Cordelia Schmid, and Andrew Zisserman, editors, *Toward Category-Level Object Recognition*, volume 4170 of *Lecture Notes in Computer Science*, pages 29–48. Springer, 2006.

[121] Arthur R. Pope and David G. Lowe. Learning appearance models for object recognition. In *In International Workshop on Object Representation for Computer Vision*, pages 201–219. Springer, 1996.

[122] T Prasad, Klaus Hartmann, Wolfgang Weihs, Seyed Ghobadi, and Arnd Sluiter. First steps in enhancing 3d vision technique using 2d/3d sensors. In *Proceedings of Computer Vision Winter Workshop CVWW'06*, pages 82–86, 2006.

[123] Gunnar Rätsch, T. Onoda, and K.-R. Müller. Soft margins for AdaBoost. *Machine Learning*, 42(3):287–320, March 2001.

[124] Gunnar Rätsch and Manfred K. Warmuth. Efficient margin maximizing with boosting. *Journal of Machine Learning Research*, 6:2131–2152, 2002.

[125] Xiaofeng Ren, , Xiaofeng Ren, and Jitendra Malik. Learning a classification model for segmentation. In *In Proc. 9th Int. Conf. Computer Vision*, pages 10–17, 2003.

[126] T. Ringbeck and B. Hagebeurker. A 3d time-of-flight camera for object detectoin. 2007. Online publication (http//www.pmdtech.com/inhalt/download/documents/070513PaperPMD.pdf).

[127] Gerhard Roth. Registering two overlapping range images. *3D Digital Imaging and Modeling*, 0:0191, 1999.

[128] Cynthia Rudin, Ingrid Daubechies, and Robert E. Schapire. The dynamics of adaboost: Cyclic behavior and convergence of margins. *J. Mach. Learn. Res.*, 5:1557–1595, 2004.

[129] Salvador Ruiz-correa, Linda G. Shapiro, and Marina Meil. A new paradigm for recognizing 3-d object shapes from range data. In *Proceedings of the IEEE Computer Society International Conference on Computer Vision 2003, Vol.2*, pages 1126–1133, 2003.

[130] Silvio Savarese and Li Fei-Fei. 3d generic object categorization, localization and pose estimation. In *International Conference on Computer Vision*, pages 1–8, 2007.

[131] Silvio Savarese and Li Fei-Fei. View synthesis for recognizing unseen poses of object classes. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 602–615, Berlin, Heidelberg, 2008. Springer-Verlag.

[132] Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: a new explanation for the effectiveness of voting methods. In *Proc. 14th International Conference on Machine Learning*, pages 322–330. Morgan Kaufmann, 1997.

[133] Robert E. Schapire and Yoram Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37:297–336, 1999.

[134] C. Schmid and R. Mohr. Combining grey value invariants with local constraints for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 872–877, 1996.

[135] Thomas Serre, Lior Wolf, and Tomaso Poggio. Object recognition with features inspired by visual cortex. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–1000, 2005.

[136] Josef Sivic, Bryan C. Russell, Alexei A. Efros, Andrew Zisserman, and William T. Freeman. Discovering objects and their localization in images. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 370–377, Washington, DC, USA, 2005. IEEE Computer Society.

[137] H. Su, M. Sun, S. Savarese, and L. Fei-Fei. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *International Conference on Computer Vision*, 2009.

[138] M. Sun, H. Su, S. Savarese, and L. Fei-Fei. A multi-view probabilistic model for 3d object classes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[139] Yiyong Sun, Joonki Paik, A. Koschan, D.L. Page, and M.A. Abidi. Point fingerprint: a new 3-d object representation scheme. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 33(4):712–717, Aug. 2003.

[140] Zehang Sun, George Bebis, and Ronald Miller. Boosting object detection using feature selection. In *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS'03)*, page 290, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[141] B. Taati and M. Greenspan. Satellite pose acquisition and tracking with variable dimensional local shape descriptors. In *EEE/RSJ IROS Workshop on Robot Vision for Space Applications*, pages 4–9, 2005.

[142] Alexander Thomas, Vittorio Ferrar, Bastian Leibe, Tinne Tuytelaars, Bernt Schiel, and Luc Van Gool. Towards multi-view object class detection. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1589–1596, Washington, DC, USA, 2006. IEEE Computer Society.

[143] J. Thureson and S. Carlsson. Appearance based qualitative image description for object class recognition. In *7th European Conference on Computer Vision ECCV04*, volume 2, pages 518–529, 2004.

[144] Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):854–869, 2007.

[145] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[146] Shimon Ullman and Ronen Basri. Recognition by linear combination of models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(10):992–1006, 1991.

[147] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (in press), 2010.

[148] Joost van de Weijer and Cordelia Schmid. Coloring local feature extraction. In *8th European Conference on Computer Vision ECCV06*, volume 2, pages 334–348, 2006.

[149] Luc J. Van Gool, Theo Moons, and Dorin Ungureanu. Affine/ photometric invariants for planar intensity patterns. In *ECCV '96: Proceedings of the 4th European Conference on Computer Vision-Volume I*, pages 642–651. Springer-Verlag, 1996.

[150] J. K. M. Vetterli. *Wavelets and Subband Coding*. Prentice Hall, 1995.

[151] Paul Viola and Michael Jones. Rapid object detection uisng a boosted cascade of simple features. In *IEEE Computer Scociety Conference on Computer Vision and Pattern Recognition CVPR01*, volume 1, pages 511–518, 2001.

[152] Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2:137–154, 2004.

[153] Paul Viola, Michael J. Jones, and Daniel Snow. Detecting pedestrians using patterns of motion and appearance. In *In ICCV*, pages 734–741, 2003.

[154] Manfred K. Warmuth, Karen Glocer, and Gunnar Rätsch. Boosting algorithms for maximizining the soft margin. *Advances in Neural Information Processing Systems (NIPS'08)*, 2008. In press.

[155] Manfred K. Warmuth, Jun Liao, and Gunnar Rätsch. Totally corrective boosting algorithms that maximize the margin. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 1001–1008, New York, NY, USA, 2006. ACM.

[156] S. Watanabe. *Pattern Recognition: Human and Mechanical*. Wiley, 1985.

[157] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *6th International Conference on Computer Vision ECCV00*, number 18–32, 2000.

[158] John Winn, A. Criminisi, and T. Minka. Object Categorization by Learned Universal Visual Dictionary. In *International Conference on Computer Vision*, volume 2, pages 1800–1807, Beijing, China, 2005.

[159] Sameh M. Yamany and Aly A. Farag. Surfacing signatures: An orientation independent free-form surface representation scheme for the purpose of objects registration and matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1105–1120, 2002.

[160] Liu Yang, Rong Jin, R. Sukthankar, and F. Jurie. Unifying discriminative visual codebook generation with classifier training for object category recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[161] Chien yuan Huang, Octavia I. Camps, and Tapas Kanungo. Object recognition using appearance-based parts and relations. In *In IEEE Conference on Computer Vision and Pattern Recognition*, pages 877–883, 1997.

[162] Wei Zhang, Bing Yu, Gregory J. Zelinsky, and Dimitris Samaras. Object Class Recognition Using Multiple Layer Boosting with Heterogeneous Features. In *IEEE Computer Society Conference on Computer Vision and Patter Recognition CVPR05*, volume 2, pages 66–73, Washington, DC, USA, 2005.

Index

[1], 117
2D/3D generic object recognition dataset, 9

AdaBoost algorithm, 20–21, 69
  Discrete, 21
  Real, 21
affine invariant interest point detector, 65
Alignment, 42

background clutter, 4
Boosting, 19–64
  preliminaries, 19
boosting, 65, 68

Caltech, 29, 47, 49, 51, 53, 54
Caltech 4, 70
CCD camera, 104
challenges, 3
CMOS photo sensors, 97
color descriptors
  local, 16
combination, 63–65
corrective, 20
counter-class, 71

dataset acquisition, 101
dense grid sampling, 138–141
descriptors
  PCA-based, 18
  regional point, 18

edge guarantee, 23
Euclidean distance, 69
expectation-maximization algorithm, 18

feature sharing, 24
filters
  Gabor, 16
  steerable, 16
functional grouping, 4
fusion, 147–161

Generalized Hough Transform, 42
generic 3D object recognition, 8
generic object recognition, 1, 2
GentleBoost algorithm, 24
GLOH, 16, 67, 82
Graz01, 30, 51
Graz02, 32, 51, 54, 70

Hessian-affine invariant detector, 66
human visual system, 1
hypothesis edge, 20

illumination variations, 4
inter-class variabilities, 6
interest point detectors, 15, 66

---

[1]

Interferometry, 97
intra-class variabilities, 5–6

joint Boosting algorithm, 24

k-means clustering, 70, 79

label noise, 91
laser beam, 98
laser scanners, 97
learning
  supervised, 18
    completely, 18
    highly, 18
    weakly, 18
  unsupervised, 18
local binary patterns, 14
local feature histograms, 18
local keypoints, 16

margin, 69
  hard, 20
  soft, 22
modulated light, 98
moment invariants, 14
monotonicity, 70

noise, 91

object categorization, 2
occlusion and truncation, 4
opponent color angle descriptor, 67–68

perceptual organization, 43
phase shift, 98
PMD camera, 96
point signatures, 17
points fingerprints, 18
probabilistic alignment, 46

radar sensors, 97
range images, 95
relative entropy, 23
representations
  2D
    local, 14–16
  global, 14
  range
    local, 16–18

shape, 95
Shape context, 43
shape descriptors, 14
SIFT, 16, 67, 82
Size and viewpoint variations, 4
soft margins, 89
SoftBoost algorithm, 21–24, 89
specific object recognition, 1
spin images, 17
stereo vision, 97
stereoscopy, 97
superpositions
  constructive, 97
  superpositions, 97
supervision, 7
support vector machines, 18
surface signatures, 17

Tensor-based correspondence, 18
Time-of-Flight
  camera, 96, 98
  principle, 97
totally corrective, 20, 22
triangulation, 97

visual class recognition, 2
visual grouping, 4

weak learner, 69
weakly supervised learning, 7, 63