Dissertation

submitted
to the
Combined Faculties for the Natural Sciences and for Mathematics
of the
Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

Put forward by
Juan Antonio Monroy Kuhn
Born in Mexico City
Oral examination: 30.10.2013

Morphological Analysis for Object Recognition, Matching, and Applications

Advisor: Prof. Dr. Björn Ommer
Prof. Dr. Christoph Schnörr

Abstract

This thesis deals with the detection and classification of objects in visual images and with the analysis of shape changes between object instances. Whereas the task of object recognition focuses on learning models which describe common properties between instances of a specific category, the analysis of the specific differences between instances is also relevant to understand the objects and the categories themselves. This research is governed by the idea that important properties for the automatic perception and understanding of objects are transmitted through their geometry or shape. Therefore, models for object recognition and shape matching are devised which exploit the geometry and properties of the objects, using as little user supervision as possible.

In order to learn object models for detection in a reliable manner, suitable object representations are required. The key idea in this work is to use a richer representation of the object shape within the object model in order to increase the description power and thus the performance of the whole system. For this purpose, we first investigate the integration of curvature information of shapes in the object model which is learned. Since natural objects intrinsically exhibit curved boundaries, an object is better described if this shape cue is integrated. This subject extends the widely used object representation based on gradient orientation histograms by incorporating a robust histogram-based description of curvature. We show that integrating this information substantially improves detection results over descriptors that solely rely upon histograms of orientated gradients.

The impact of using richer shape representations for object recognition is further investigated through a novel method which goes beyond traditional bounding-box representations for objects. Visual recognition requires learning object models from training data. Commonly, training samples are annotated by marking only the bounding-box of objects since this appears to be the best trade-off between labeling information and effectiveness. However, objects are typically not box-shaped. Thus, the usual parametrization of objects using a bounding box seems inappropriate since such a box contains a significant amount of background clutter. Therefore, the presented approach learns object models for detection while simultaneously learning to segregate objects from clutter and extracting their overall shape, *without* however, requiring manual segmentation of the training samples.

Shape equivalence is another interesting property related to shape. It refers to the ability of perceiving two distinct objects as having the same or similar shape. This thesis also explores the usage of this ability to detect objects in unsupervised scenarios, that is where no annotation of training data is available for learning a statistical model. For this purpose, a dataset of historical Chinese cartoons drawn during the Cultural Revolution and immediately thereafter is analyzed. Relevant objects in this dataset are emphasized through annuli of light rays. The idea of our method is to consider the different annuli as shape equivalent objects, that is, as objects sharing the same shape and devise a method to detect them. Thereafter, it is possible to indirectly infer the position, size and scale of the emphasized objects using the annuli detections.

Not only commonalities among objects, but also the specific differences between them are perceived by a visual system. These differences can be understood through the analysis of how objects and their shape change. For this reason, this thesis also develops a novel methodology for analyzing the shape deformation between a single pair of images under missing correspondences. The key observation is that objects cannot deform arbitrarily, but rather the deformation itself follows the geometry and constraints imposed by the object itself. We describe the overall complex object deformation using a piecewise linear model. Thereby, we are able to identify each of the parts in the shape which share the

same deformation. Thus, we are able to understand how an object and its parts were transformed. A remarkable property of the algorithm is the ability to automatically estimate the model complexity according to the overall complexity of the shape deformation. Specifically, the introduced methodology is used to analyze the deformation between original instances and reproductions of artworks. The nature of the analyzed alterations ranges from deliberate modifications by the artist to geometrical errors accumulated during the reproduction process of the image. The usage of this method within this application shows how productive the interaction between computer vision and the field of the humanities is. The goal is not to supplant human expertise, but to enhance and deepen connoisseurship about a given problem.

Zusammenfassung

Diese vorgelegte Dissertation befasst sich mit der Ekennung und Klassifizierung von Objekten in Bildern und mit der Analyse von Formveränderungen zwischen Objekten. Während Objekterkennung sich mit dem Lernen von Objektmodellen befasst, die die Gemeinsamkeiten zwischen Objektinstanzen beschreiben, ist die Analyze von spezifischen Unterschieden zwischen Objektinstanzen nötig, um die Objekte und Kategorien selber zu verstehen. Die Leithypothese dieser Forschung ist, dass wichtigsten Eigenschaften für die vollautomatische Perzeption und das Verstehen von Objekten durch ihre Form oder Geometrie gegeben sind. Folglich werden in dieser Arbeit Modelle für Objekterkennung und *Form-Matching* entwickelt, die die Formeigenschaften von Objekten mit möglichst wenig Überwachungsinformation verwenden.

Um zuverlässige Objektmodelle zu lernen, werden angemessene Objektdarstellungen benötigt. Die Idee dieser Arbeit liegt darin eine genauere Beschreibung der Objektform[1] zu verwenden, die die Beschreibungsmöglichkeit des Objektmodels selber und somit auch die Performance des gesamten Systems erhöht. Für diesen Zweck untersucht diese Arbeit zunächst die Integration von Krümmungsinformation der Objektform in dem zu lernenden Objektmodell. Da natürliche Objekte intrinsisch eine gekrümmte Form aufweisen, sollte das Objektmodell die Krümmungsinformation integrieren. Die vorliegende Arbeit erweitert die weitverbreitete, auf Orientierung von Gradienten basierte Objektbeschreibung durch die Einfügung einer robusten, histogram-basierten Beschreibung der Krümmung. Durch Verwendung dieser komplementären Information kann das Erkennungsresultat substantiell verbessert werden.

Im Weiteren werden durch eine neue Methode die Auswirkung der Verwendung der Objektgeometrie für Objekerkennung untersucht, die über die gewöhnliche Methode der auf Bounding-box basierten Objektdarstellungen hinausgeht. Die Visuelle Erkennung von Objekten erlernt Objektmodelle mit Hilfe von Trainingsinformationen. Im Allgemeinen werden die Objekte innerhalb solcher Trainingsbeispiele mit einer Bounding-box markiert, da dies den besten Ausgleich zwischen manueller Beschriftung und Effektivität zu sein schien. Allerdings haben Objekte keine Boxform, sodass die gewöhnliche Objektbeschreibung durch Lage, Skala und Askpektverhältnis nur unzureichend widergegeben wurde. Der Grund dafür ist, dass die Box selbst viele Hintegrundsstördaten beinhaltete. Im Gegensatz dazu stellt die vorliegende Arbeit eine Methode zum Erlernen von Objektmodellen vor, bei der gleichzeitig sowohl die Abgrenzung von Objekten zu ihrem Hintergrund als auch die Erzeugung der gesamten Objektform erlernt wird. Dies geschieht ohne manuelle Segmentierung der Trainingsbespiele.

Formäquivalenz ist eine weitere interessante Fähigkeit, die in Beziehung zu der Geometrie eines Objektes steht. Sie beschreibt die Fähigkeit ähnliche Objektformen zwischen verschiedenen Objekten wahrzunehmen. Diese Dissertation erforscht ihre Verwendung im Bereich der nicht überwachten Objekerkennung, d.h. der Objekterkennung, bei der die Annotation der Trainingsbeispiele für das Lernen eines statistischen Modeles entbehrlich ist. Zu diesem Zweck wird eine nicht annotierte Datenbank von chinesischen Comicbildern analysiert, die in der chinesischen Kulturrevolution entstanden sind. Für den Autor des Comics wichtige Objekte werden in diesem Datensatz mit Hilfe von ringförmigen Lichtstrahlen hervorgehoben. Die Idee dieser Methode besteht darin, die verschiedenen ringförmigen Kränze als formäquivalente Objekte zu betrachten, d.h. als Objekte mit einer gleichen Form, und eine Methode für ihre Erkennung zu entwickeln. Mit Hilfe der erkannten Lichtstrahlen, ist es möglich die Lage, Größe und Skala der hervorgehobenen

[1] Object shape

Objekte innerhalb des Comics abzuleiten.
Nicht nur Gemeinsamkeiten sondern auch spezifische Unterschiede zwischen sich ähnelnden Objekten werden von einem visuellen System wahrgenommen. Diese feinen Unterschiede können durch die Analyse der Veränderung der jeweiligen Objekteformen verstanden werden. Aus diesem Grund entwickelt die vorliegende Arbeit eine neue Methode, um die Formveränderungen zwischen zwei Bildern zu beschreiben, zu quantifizieren und gleichzeitig die Korrespondenzen zwischen den Objekten zu finden. Die entscheidende Erkenntnis ist, dass Objekte nicht beliebig deformierbar sind, sondern jede Deformation der Geometrie und ihrern Nebenbedingungen entsprechen muss. Die komplexe Gesamtdeformation eines Objektes wird mit Hilfe eines stückweisen linearen Modelles beschrieben. Dadurch können die verschieden Teile der Geometrie erkannt werden, die in einem zusammenhang transformiert wurden. Diese Gruppierungen ermöglicht die Visualisierung und das Verständnis der gesamten Objekttransformation. Eine wichtige Eigenschaft des Algorithmus ist die Möglichkeit, die Modellkomplexität (d.h. die Anzahl der nötigen linearen Transformationen für die Registrierung der Objekte) automatisch entspechend der zugrundeliegenden Deformation zu bestimmen. Das Modell wird verwendet um subtile Änderungen zwischen einem Originalkunstwerk und dessen Reproduktionen zu analysieren. Die Natur der Bilddeformationen variiert von absichtlichen Abänderungen von Seiten des Künstlers bis zu geometrischen Fehlern, die während des Reproduktionsprozesses aufgetreten sind. Diese Anwendung zeigt zugleich, wie gewinnbringend die Interaktion zwischen Computer Vision und Geisteswissenschaften sein kann. Das Ziel besteht nicht darin menschliche Kompetenz zu ersetzen, sondern das Verständnis einer Objektentwicklung zu vertiefen und genauer zu formulieren.

Acknowledgements

My sincere thanks go to all the people that have contributed in various ways to the successful completion of this thesis and my research work over the past years.

First, I would like to thank my advisor Björn Ommer for introducing me to the research of this fascinating field and for providing a fruitful research environment in the group. My sincere thanks go also to Professor Christoph Schnörr for for his feedback regarding this thesis.

Thanks go also to my colleagues in the HCI and especially in the computer vision group: Angela, Pradeep, Boris, Peter, Bernd, Masatto and Jose for the interesting joint work and discussions about computer science. I also had a lot of fun discussing other important topics in life. There is a nice saying in Germany for these kind of discussions: *"über Gott und die Welt reden"*. I am also grateful for Tanja Kohl's help in administrative matters. Also Ole Hansen, Juergen Moldenhauer and Markus Nullmeier did a great job at keeping the computing services in excellent shape.

It would be very difficult to list all the people that supported me and encouraged me over the past years. A special thanks goes to Thomas, Bruno, Johanna and Jenny. Finally, special thanks go to Maren and my family for their love, encouragement and all the happy moments we spent together.

CONTENTS

1 Introduction **1**

1.1 Modeling Reality 1

1.2 Computer Vision 2

1.3 Excursus: The Philosophical Foundations of Computer Vision 4

1.3.1 Knowledge as a Result of Sensory Perception 4

1.3.2 Using Machines to Understand Perception 6

1.4 The Importance of Shape 6

1.5 Open Questions in Computer Vision 9

1.5.1 Simplified object representations in object recognition systems . . . 9

1.5.2 Modeling Shape Changes 11

1.6 Objectives of the Present Thesis 13

1.7 Original Contributions 14

1.8 Organization of the Thesis 15

2 Using Shape for Object Recognition **17**

2.1 Local Image Descriptors 17

2.2 General-Modeling Paradigms 20

2.3 Shape-Modeling Paradigms 23

2.3.1 Indirect vs Direct Shape Representation 24

2.3.2 Contour-based vs Region-based Shape Representation 25

2.3.3 Supervision Degree 26

3 Beyond Straight Lines - Object Detection Using Curvature **29**

3.1 Using Curvature for Object Recognition 29

3.2 The Contribution 30

3.3 Curvature Estimation Methods 31

3.3.1 Curvature Cues for Object Detection 32

3.4 Robust Representation of Curvature 32

3.5 Model Learning 33

3.5.1 Support Vector Machine (SVM) 33

3.5.2 Intersection Kernel SVM to Learn Curvature Representation 36

3.6 Experimental Results 37

3.7 Discussion 38

4 Beyond Bounding-Boxes: Learning Object Shape by Model-driven Grouping 43
4.1 Going Beyond Bounding-Boxes 43
4.2 Novelty of the Approach 44
4.3 Suppressing the Background 45
4.4 Matching Objects 46
4.5 Learning from Unsegmented Training Data 47
4.5.1 Multiple Instance Learning (MIL) paradigm 47
4.5.2 Our Model: Using MIL to Find Foreground Masks 49
4.6 Data-driven Grouping 51
4.7 Learning a Prototypical Set of Segments 51
4.8 Implementation Details 52
4.9 Experimental results 52
4.10 Discussion 56

5 Beyond Annotated Datasets - Parametric Object Detection for Iconographic Analysis using Shape Equivalence 59
5.1 Parametric Object Detection for Iconographic Analysis 59
5.2 Using Shape Equivalence for Object Detection 60
5.3 Edge Extraction 61
5.4 Line Fitting and Clustering 61
5.5 Object Localization and Detection 64
5.6 Results 65
5.7 Discussion 65

6 Describing Shape Changes 67
6.1 Shape Changes Under Given Correspondences 67
6.2 Shape Change Under Missing Correspondences 71
6.3 Limitations of Current Approaches 76

7 Beyond Global Transformations - Morphological Analysis for Investigating Artistic Reproductions 79
7.1 Introduction 79
7.2 Novelty of the Approach 82
7.3 Automatic Estimation of Transformations, Groups and Correspondences . . 86
7.3.1 Problem Statement 86
7.3.2 Optimization Strategy 88
7.3.3 Choosing the Right Number of Clusters 92
7.4 Experiments 93
7.4.1 Synthetic data 93
7.4.2 Reproductions of the Codex Manesse 95
7.4.3 Michelangelo Reproductions 99
7.4.4 Implementation Details 99
7.5 Discussion 100

8 Conclusions 103

Bibliography 107

CHAPTER 1

INTRODUCTION

1.1 Modeling Reality

When Copernicus, Tyco Brahe and Galileo developed a more accurate picture of the solar system as being heliocentric, only then were many of Aristotle's theories about heavenly bodies proven false ([102] p. 274-275). After Christopher Columbus discovered the new world, only then did the perception about the physical world radically change. It is the interaction between observing nature and finding a coherent explanation for these observations that generates knowledge. To know which relations exist between observable complex real-world facts and being able to express these observations produces scientific knowledge. It can be stated that science has helped humanity to understand its surroundings and to understand itself as part of nature. Ultimately, the advances in technology we have experienced in the last two centuries are a product of this process of knowledge generation. It can be further stated that technology emerges when man intends to use knowledge in order to solve problems which affect his living in the world. Therefore, scientific and technological understanding of a process are intimately related to the formulation of functional relation between a representation or model and the real-processes we observe in nature. The thermostat is a good example to explain this fact. There exists a functional relation between the temperature in the external world and the temperature which is represented by the height of the metallic strip in the thermostat, we *know* about the outside-world temperature by looking at the thermostat. A more abstract example is the relation between an apple which falls from a tree and the universal theory of gravitation formulated by Isaac Newton. We are able to explain and know why such an amazing fact like the interaction of gravity with an apple happens by making use of a theory. This idea of knowledge generation can be embedded in an information-processing framework. According to this framework, there exists a representational system which links two related but distinct worlds. On the one side we have the external world and on the other side we have the *representing world* which contains models or representations of the external world ([196]). In this framework, scientific knowledge begins with the generation of models in the representing world which are related to observations in the external world. In a second step, these models are applied to the external

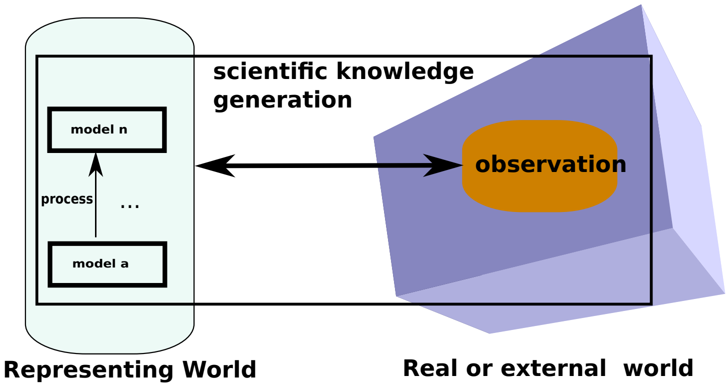


Figure 1.1: Generation of scientific knowledge as described in section 1.1.

world to make new observations, which deny, corroborate, improve or show the capability of a given model for describing reality. From this perspective, to know something consists of knowing the homomorphism that links a model (or group of models) and the external world. In some cases, due to the complexity of reality, it is required for a description to use different models which need to be linked. Therefore, the term *process* is used to describe the mapping from one representation or model to another ([170]). A graphic illustration of this understanding of knowledge generation can be appreciated in figure 1.1.

The history of modern science can be understood as a history of struggling for better and more precise models which describe and represent reality in such a manner that we can better predict the behavior of nature. We are able to predict the falling of an apple by using the Newtonian framework, but we are able to express this fact and describe a broader scope of reality if we use the more powerful and complete framework conceived by Einstein. Furthermore, a good scientist should always remember that both worlds are related but not identical: a model is a representation of reality but not reality itself. Only out of this motivation can science develop in its attempt of obtaining better models.

The understanding of human perception and the visual system has not been any exception to scientific undertakings. Much research has been done and many theories of how humans see and perceive the real-world have been formulated. It is unquestionable that research within this field broke new ground with the introduction and developments of modern computers during the 50's and 60's (e.g. see [170]), but it was the effort of scientists intending for the first time to implement and simulate the different vision theories when a new field in science arose: Computer Vision.

1.2 Computer Vision

The original goal of Computer Vision was to devise algorithms that enable computers to understand the visually perceivable world ([193]). Nevertheless, during the 70's, when scientists started using computers to address vision problems (e.g. the group around Marvin Minsky is a good example), it became clearer that simulating human vision is a far more complicated task than they had imagined. All theoretical and computation models at that time were incapable of solving problems, which until then has been considered trivial by human vision theories. The most representative example is probably the problem of extracting the edge-signal out of images. As David Marr formulated in his very influential book *Vision* [170], many vision theories agreed upon the fact that human vision is capable of extracting edges from the retinal image at an early stage of vision. However, the available computer methods at that time were only partially capable of extracting edge-signals from images since the solution of this task was limited to very controlled scenarios where images did not present any noise factors such as changes in illumination, boundary occlusions, or background clutter. Only after more than 30 years, with the work of Fowlkes et al. [173], computer scientists are now capable of extracting edges under real-world scenarios, if not perfectly, at least in a reliable manner.

This discrepancy between human vision theories and computational feasibility pointed out by David Marr has since then led on the one hand to a fragmentation of the computer vision research field into different sub-disciplines which we can observe nowadays. These sub-disciplines include methods for acquiring, analyzing and understanding images [121] and commonly are studied without any global “vision theory” capable of integrating the generated knowledge into a single system. However, on the other hand, this discrepancy itself together with the complexity and highly diversified functionality of the human visual system, the most perfect vision system we have access to, has kept alive the greatest goal of computer vision: developing a vision algorithm capable of passing the Turing-Test, which means that the machine’s ability to see should be equivalent to, or indistinguishable from, that of an actual human.

An analogous exemplification from history of knowledge which helps to understand the relation between the human visual system functionality and practical research in computer vision today is the invention of airplanes. Birds have fascinated men for many centuries due to their ability to fly. In fact the study of birds and their flying was the starting point for building the first flying devices. For instance, Leonardo Da Vinci research about the wings of birds for his designs of an aircraft in his *Codex on the Flight of Birds* (1502) is an early example of this. However, it wasn’t until 1903 that the Wright brothers developed the first powered airplane with sustained flight. Nowadays, despite existent similarities (e.g. the usage and control of wings and general aerodynamic principles) between the manner in which birds fly and plane flight, crucial differences between them like the source of propulsion are also evident. Therefore, it cannot be stated that modern flying devices simulate the flight of birds. However, they cannot be understood without noting the first inspiration and endless studies of the flying of birds over time. The reason for this is that every advance in knowledge always presupposes the analysis of previously existing state-of-the-art systems. This fact can also be described by the image of Bernard of Chartres which states that we always *stand on the shoulders of giants.*[1]

Visual perception is intrinsically related to objects. To the question: “What do I see?”, the answer will most often be: objects or events among which objects play the central role. Therefore, it is reasonable to understand visual perception as the process of acquiring knowledge about objects and events by extracting information from the light they emit or reflect ([196], s. 5). As a matter of fact, the human mind does not only perceive objects as having a particular shape, color, and position, but the mind is also capable of recognizing or identifying an object as belonging to a certain group of objects which share common characteristics (which we commonly call classes) and also perceiving the differences between the instances of certain classes. The process of classification allows humans to gather all required information to be able to interact with an object in an

[1] This sentence has been attributed to Bernard of Chartres by John of Salisbury in 1159 within his opus Metalogicon

appropriate way. Additionally, once the class is known, the mind is able to establish the functionality of an object: Once I recognize an object as a chair, I will be able to sit upon it. Therefore, *detecting and classifying objects* within an image is a key task in the development of automatic vision systems in computer vision.

Furthermore, once objects are detected, the human visual system is capable of relating the objects to other objects in the scene or to previously seen objects which are present in the memory and to recognize subtle variations between them. The nature of these variations is very diverse. For instance, differences in color are easily detected, but also differences in size and shape can be perceived. Therefore, whereas the task of classification requires to abstract the commonalities between different objects in order to recognize and study unique instances of classes, an analysis of the specific differences between objects is also necessary to truly understand *what* an object is. The book of D'Arcy Thompson [231] *On Growth and Form* is an excellent example of how useful shape is for this task. Thompson explained species differences considering deformations between the different shapes of class members. This was done by drawing a regular square grid on one object and deforming it until it lay on a second grid of the other object, with corresponding biological parts located in the corresponding blocks.

Therefore, a successful computer vision system requires at least two tasks to be solved. The first task is the recognition and classification of objects and the second will be the analysis of differences between the shapes of different objects. Both tasks are studied in this thesis.

1.3 Excursus: The Philosophical Foundations of Computer Vision

In this section the ideas of section 1.1 and section 1.2 are further pursued in a rather unusual manner. In the last section it became clear that the original goal of computer vision was and is to devise algorithms that enable computers (machines) to understand the visually perceivable world. Now, in this section we analyze from a philosophical point of view, *what is required to be thought* by a human in order to pursue the original goal of computer vision. In other words, we are interested in finding and sketching the historical genealogy of the ideas that made thinking the new idea of computer vision possible. This analysis is done in order to reflect on the foundations of the field and to realize that its roots lie far before the first computers were invented. Furthermore, not a single idea, but rather a complex combination of philosophical results and thoughts through history were required to make it possible for humans to think about computer vision. Due to the complexity of the aforementioned task, we limit ourselves to sketch (in a rather crude manner) two important ideas which seem to lay the foundations of computer vision

1.3.1 Knowledge as a Result of Sensory Perception

It is the intention of computer vision to extract or infer information from sensory data (e.g. visual images) in order to generate knowledge or understanding about the underlying pictured reality. However, the idea of using sensory perception by humans to generate knowledge has not been self-evident in the history of human knowledge.

In ancient philosophy, Socrates expressed a belief that the material world as it seems to us is not the real world, but only an image or copy of the real world. This belief was formulated by Plato in his very influential allegory of the cave [200]. According to this allegory, mankind can be compared to people who have lived chained to the wall of a cave all of their lives facing a wall. Furthermore, those people can only watch the shadows projected on the wall by the things that pass in front of a fire behind them. Watching these shadows is the only manner of getting to view reality. Therefore, philosophical knowledge is the only way to get free of the chains in order to see the *real world.* Using this allegory, Socrates concluded that current human perception was not directly related to how the world was in itself but rather that sensory perceptions are only mappings of eternal ideas and thus they should not be relevant to the generation of knowledge, which can only be reached by philosophical thinking. Socrates considered the whole natural universe as an epiphany, that is, as an image of divinity [201] or divine ideas.

This idea that reality is a mapping of divinity was widely accepted in Arabic and Latin metaphysics in later centuries and thus human thinking concentrated itself on understanding the divinity and its ideas (that is, theology was developed) in order to understand the *real* world. However, during the middle ages several philosophers and theologians under the influence of Aristotle and later Arabic scholars started thinking about human knowledge not only as a mapping of the eternal, but as a mapping of a finite reality. According to this idea, nothing can get into the human mind if it is not captured before by the human senses. Within this thinking, it was Thomas Aquinas (1225-1274) who postulated the idea that *knowledge or truth* is based upon the agreement or concordance between the intellect and the *real thing* (*adequatio rei et intellectus*). Furthermore, the Franciscan friar Roger Bacon (1214-1294) represents this new idea which emerged during the middle ages best and looked at experience and sensory perception not only as symbols for another reality, but also asked for its own structure and nature laws. And in fact, Heimsoeth in his very influential book [112] considers Roger Bacon to be a very important thinker of the middle ages who helped us to understand the philosophy developed during the Renaissance, which builds the fundamentals of the later *empiricism* of the 17th century. And indeed, it was during the Renaissance when the thinking that a real outside world exists and humans have senses to perceive it deepened. For instance, the human perception organs (e.g. the eyes) generate sensations; images of the world in our consciousness. Hence, humans are compelled to develop better instruments to generate better images of reality in our consciousness (e.g. thermometers, barometers, telescopes, microscopes). And for sure, this new understanding of nature can also be seen as the underlying motive that inspired Galileo Galilei to express (e.g. in [**?**]) that nature "... is written in the language of mathematics".

This thinking of the Renaissance was absorbed during the 17th century in the English empiricism by postulating that *everything* that humans know is only possible through human sense perception (e.g. [155]). Sensory perception became the only source of knowledge. Furthermore, this school of thought considered the human soul as a *tabula rassa*, an empty blackboard that through sensory perception of reality becomes substantiated. For instance, in his *Essay concerning Humane Understanding* John Locke described the mind as an "empty cabinet" or "waxed tablet". And indeed, this way of thinking and its further reception have had an important influence on the development of modern science [23]. Moreover, it is this idea of understanding reality and knowledge generation based on sensory information that also lies behind computer vision, where a full understanding of a scene is intended using only sensory imagery.

1.3.2 Using Machines to Understand Perception

Computer vision, as described in section 1.2, is a field which has emerged from the attempts to understand and simulate the human vision system with the help of computers. And in fact, new discoveries about the functionality of the brain have triggered this research. For instance, the work of Louis Lapicque about the *integrate and fire* model of the neuron can be considered as an early attempt to understand the brain. Another example is the work of Hubel and Wiesel [118], who discovered that neurons in the primary visual cortex (V1) react to oriented stimuli. It was along this line of research that David Marr used a computational theory in order to explain the interaction and information process between neurons or groups of neurons (e.g. [169, 168]). From this specific perspective the brain's functionality started to be thought of as being like a machine, and thus the usage of machines to simulate brain activity became plausible. However, although the novelty of this idea and its plausibility relied on new observable scientific evidence, this idea can be seen in the history of a wider stream of thinking.

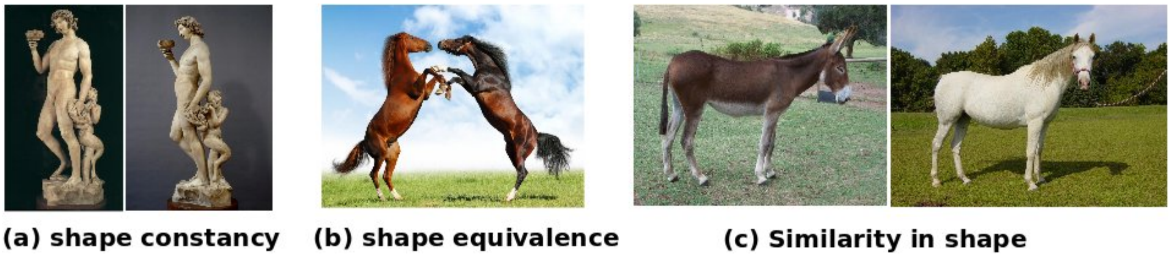
Already ancient philosophies like the atomists or epicurean philosophy believed that the universe could be fully explained by mechanical principles acting upon atoms [209]. These ideas were further developed by the early *mechanical philosophy* of matter during the early modern period [62]. According to this belief, living things can be understood as machines. And indeed, many achievements during the scientific revolution showed that many phenomena could be explained in terms of "mechanical" laws or natural laws that act upon matter. For instance, R. Descartes understood animals and humans as mechanistic automata. For instance, in his work "Treatise of Man" (p. 108) he wrote:

> "I should like you to consider that these functions (including passion, memory, and imagination) follow from the mere arrangement of the machine's organs every bit as naturally as the movements of a clock or other automaton follow from the arrangement of its counter-weights and wheels."

Nevertheless, R. Descartes explained only vital functions and automatic actions (e.g. habits) in terms of mechanistic interactions of matter. On the contrary, activities like conceptual thinking and free will were understood as purely mind activities (s. [50] p. 60-61). T. Hobbes (1588-1679), in contrast, conceived the human mind as purely materialist-mechanistic (see his work [115] published in 1651), fully explicable in terms of the effects of sensory perception, which in turn is explained by the operations of the nervous system. (s. [50] p. 102-103). This was a new idea that probably was not possible to reach due to the lack of advanced technology at that time. However, as mentioned above, recent scientists like David Marr, inspired by new physical discoveries and observations about the human brain and specifically about human vision, retook similar ideas as Hobbs and his school of thinking and by doing this, laid an important thinking paradigm for modern computer vision.

1.4 The Importance of Shape

It is hardly imaginable to conceive of a powerful vision system incapable of distinguishing or recognizing shape. This becomes clear considering a retinal disease, where the patient is only capable of vaguely perceiving the color but is unable to sharply see the contours of the objects. A severe Stagardt's disease is such an example, where a gradual degeneration



(a) shape constancy (b) shape equivalence (c) Similarity in shape

Figure 1.2: The human vision system is capable of solving three tasks with respect to the shape of an object. (a) Two different shapes belong to the same sculpture (Bacchus by Michelangelo, 1496/97; Florence, Museo Nazionale del Bargello) (b) Two different objects feature the same (or similar) shape (c) The differences in shape between different objects can be perceived.

of the macula is produced. The macula is the area in the middle of the retina that makes the central vision needed for daily life activities possible. Its degeneration leads to a loss of detailed vision, thus strongly limiting the perception of many properties which characterize an object.

Shape is probably the most important and most complex to describe property we perceive about objects [196]. One reason for this is that through shape we implicitly perceive all other spatial properties like size, orientation and position. Therefore, shape becomes crucial for the human vision system to solve distinct tasks like determining the category and function of an object [196]. In fact Biderman et al. ([21]) considers shape as the most important type of information required for the categorization of objects. However, the human vision system not only limits itself to recognizing and categorizing objects, but through shape it is also able to perceive similarities and differences between objects. The human vision system is capable of solving (at least) three tasks with respect to the shape of an object [196]:

- *Shape constancy* is the capability of perceiving that two shapes belong to the same object regardless of the difference in viewpoint
- *Shape equivalence* is the capability that refers to the fact that humans are able to perceive two distinct objects as having the same shape
- *Similarity in shape* refers to the ability of perceiving the differences in shape between different objects.

The first two capabilities are closely related since shape constancy is defined regarding the *same* object and shape equivalence is concerned with the relation between two *different objects*. The ability for detecting shape equivalence is illustrated very well when a human is confronted with an unknown shape. Although he has no prior knowledge about the object, he possesses the ability of detecting other objects which share the uncommon shape. For example, in chapter 5 we will make direct use of this property in order to carry out an unsupervised iconographic analysis of Chinese cartoons, drawn during the second half of the Cultural Revolution and immediately afterwards.

The third task, detecting similarity in shape, is probably the most interesting one, since regardless whether the differences are big or subtle, humans are capable of recognizing two shapes as similar but not identical. Moreover, humans are capable of localizing the differences within the shapes and quantifying them. For instance, if a human is able to

recognize two tables, he is also able to describe that the legs (that is, he is able to localize the deformation) of one desk are straight, whereas the legs of the second table are more curved (that is, he quantifies the deformation).
Moreover detecting similarities in shape is a difficult problem from the mathematical point of view, because it requires establishing a framework in which the differences can be localized and described at the same time. A deeper understanding of this mathematical modeling framework is given in chapter 6.
Since the perception of the differences in shape is crucial to infer properties or to understand the anatomy of the object itself, the analysis of shape has found several applications in different fields as:

- In *Biology*: For instance, Drucker et al. [70] described an investigation to discover the cranial differences between the sexes of apes, where it is of interest to understand whether there is a size difference between the sexes and in this case investigate whether there exist shape differences in the face and braincase regions.

- In *Medicine*: Bookstein [24] for instance, analyzed brain scans of schizophrenic patients and normal patients in order to study the shape differences of the brain between the two groups.

- In *Cultural Heritage*: European Art of the Middle Ages and The Early Modern period were mainly reproduced in black and white prints or monochrome drawings which were also used to prepare paintings, sculptures, architecture or tapestries (e.g. [180]). Humans are able to distinguish the differences between the shapes in the preliminary studies and the final artworks and to recognize that both shapes belong to the same object in spite of the differences in shape. This example is relevant since it builds the starting point for the development of a new shape analysis model in chapter 7.

The importance of using shape for visual perception tasks has also been identified early in the field of computer vision. Although the concept of shape has been formalized in different manners, the term has been used mainly to refer to the spatial structure or global geometry of an object (e.g. [193] [129] [26], [70], [226]). The most common definition of shape was formulated by Kendall [130]:

> “Shape is all the geometrical information that remains when location, scale and rotational effects are filtered out from an object”

This definition is interesting since it can be considered as a negative definition. It only enlists all elements that do *not* belong to shape (i.e. location, scale and rotation) but does not explicitly specify the understanding of “geometrical information”. The fact that shape is given in terms of a negative definition is clear evidence of the complexity of this object property. Furthermore, this evidence gets hardened by observing all different models that have been used in computer vision to describe shape and use it to solve different tasks like object recognition, shape matching or shape analysis. For instance, the usage of shape for automatic object detection is described in chapter 2.

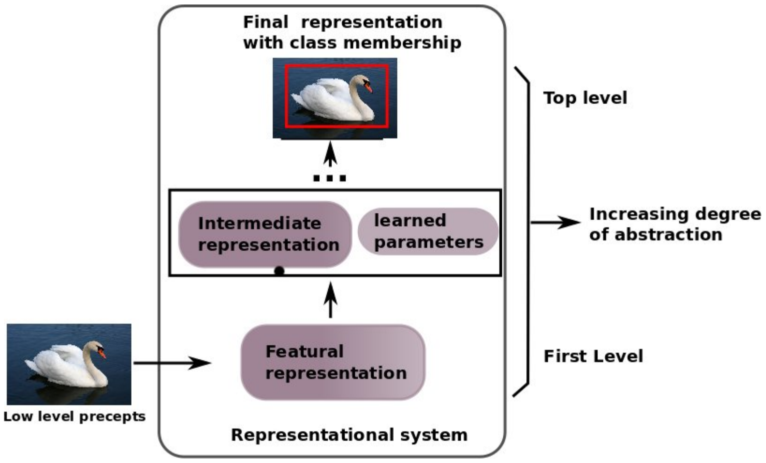


Figure 1.3: Object Recognition systems can be considered as representational systems that map initial measurements of an image into the space of object representations and class assignments (s. section 1.5.1).

1.5 Open Questions in Computer Vision

Among the diverse and numerous challenges which computer vision still requires to solve in order to get closer to the final task of developing a unified algorithm capable of truly "seeing" and understanding the visual content of an image, the observations in previous sections suggests that there exist two problems that lie at the heart of computer vision. The goal of the current thesis is to study and devise new methods which focus on these challenges. However, before giving a detailed description of the goal and original contributions of the current thesis, we briefly describe these challenges.

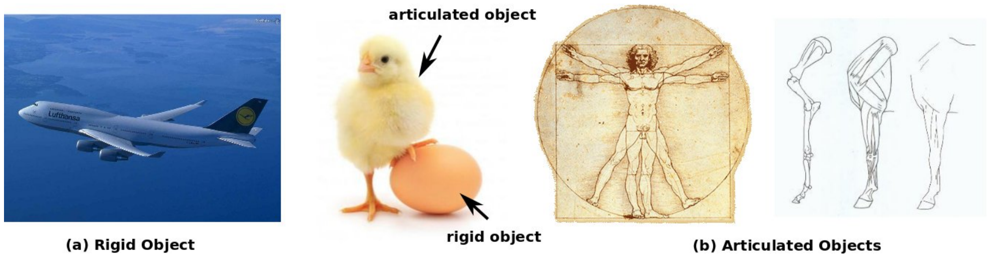
1.5.1 Simplified object representations in object recognition systems

Object Recognition Systems

Automatic object recognition is the activity of detection and classification of novel object instances as belonging to a certain class using computers. For this purpose, a class representation needs to be "learned" by the system from a limited set of training images, and during detection the system requires to find an object and decide its class membership based on the learned parameters. From this point of view, an object recognition system maps initial measurements of an image into the space of object representations and class assignments. The starting point of this process is an initial abstraction of the low level percepts which corresponds to the pixels of a digital camera (or in case of the human visual system to photoreceptive cells in the retina) [193]. Together with the learned class parameters, this abstraction process ends in turn with the final representation of the object (together with its class assignment). Normally, this last representation is given by a bounding box containing the query object. In other words, such systems output a four-parameter representation of the object: position, scale, size of the box and class membership. Although further parameters could be specified (e.g. rotation), the object itself is thought of as being a rectangular box. The purpose of such an abstract, rather simple representation consists in being semantically accessible to a human interactor, since a two dimensional bounding box is more understandable than a vector containing the raw color values of the pixels belonging to the object. Therefore, each automatic object recognition system can also be understood as the process of closing the semantic difference or *semantic gap* between the initial and final representation of an object of a certain class. The modeling choices made by different recognition systems become equivalent to choosing different ways and methods of bridging this semantic gap and will be discussed in chapter 2. For the moment it is enough to consider the overall object recognition system as a representational system that performs an abstraction process consisting of different levels. A graphical schema of such a system can be observed in figure 1.3.

Simplified Object Representations

Since each object recognition system is based upon an abstraction process, the success of a system is closely related to the capacity of description at each level by the underlying model. That is, the more information about the object that it is lost at lower levels of the representational system, the less information that can be used at higher levels to solve the final problem of finding the object understanding the visual content of the image. Thus, object representations at different levels that are too simple will limit the performance of the overall system. Two examples may illustrate this problem: On the first level of this abstraction process a feature-based representation or local image description of the object is commonly found. For instance, the very common representation based on histograms of oriented gradients (HoG, [60]) is a good example. In this description, the image region containing the object is subdivided into a regular grid. Thereafter, in each cell of the grid the orientation of the gradient signal is discretized and weighted with its magnitude in order to build a histogram which builds the intermediate representation of the image content of that cell. The local intermediate object description consists then of the concatenation of all cell histograms. This rather simple representation results in a local straight line approximation of object boundaries since local regions are described by a histogram over a discrete set of the edge orientations that they contain. In this framework smooth curves cannot be distinguished from sharp bent curves and thus valuable information gets lost at this level. The next level will then operate with a poor representation that loses the valuable information about the curvature of the object. In this example it is clear how the *description of the object's shape* in a model directly influences the mapping from the real-world object to the space of object representations. Such influence, as we will show in this thesis, has an important impact on the performance of an object detection system. A similar problem occurs in systems where objects are represented by box-shaped templates. The first problem which becomes evident is that in this case, the shape of an object is not box-shaped and so the detection window contains significant amounts of background clutter that tend to deteriorate the decision about the class identity of the present object. Secondly, the object shape can only be used for detection purposes only after being segregated from the background. Therefore, a rather poor and rude abstraction of the shape of the object (if a box is considered as the shape) used by the underlying model will limit the performance of the system due to the description capacity of the abstraction itself. Furthermore, if the final task of a computer vision system consists of understanding the content of the image, and therefore understanding the interaction of an object with its surroundings, it becomes arguable whether this task is fulfilled using a bounding-box representation of the object that is not capable of distinguishing the object from its background.



(a) Rigid Object

(b) Articulated Objects

Figure 1.4: Rigid vs articulated objects (s. section 1.5.2). A *rigid body* is defined as an object, where all of its particles (or points on it) maintain approximately the same distance relative to each other through time or in comparison with another similar object.

1.5.2 Modeling Shape Changes

It is evident that the human visual system does not limit itself to detect, classify or characterize objects through global properties like size, orientation and position. Moreover it is able to "understand" the object it recognizes not only by means of positioning a bounding-box each time an object is detected. For sure, the task of object recognition is a crucial step towards this general "understanding" about objects since it is first required to know *where* an object is in order to understand it. However, once objects are detected, a deeper analysis of the structure of the object itself is advised.

For instance, the degree of rigidity of an object is an important characteristic which reveals the structure of the object itself. In everyday life the rigidity of an object is perceived through the nature of how it moves or comparing the differences with respect to another instance of the same object class. This perception is best explained if a *rigid body* is defined as an object where all of its particles (or points on it) maintain approximately the same distance relative to each other through time or compared with another similar object. Mathematically this means that given two objects of the same class, the object is perceived as a rigid object if the transformation between instances can be described using a mapping that includes scaling, rotation, and translation, since these kinds of transformations precisely maintain the distances between every pair of points in a vector space. Whereas an egg is perceived by the human mind as a rigid object, the human body is considered as an *articulated object.* Analogously, an articulated object is where there exist at least a pair of points on the shape which change their relative distance through time or in comparison with another similar object. Mathematically, the global transformation between articulated objects has normally a non-linear character. However, there is a further observation which is helpful to specify the nature of this transformation. The deformation of natural articulated objects is not arbitrary, but rather every articulation can be described or approximated by means of local rigid or affine transformation. For instance, whereas the movement of a single leg of an animal (e.g. see figure 1.4) is nonlinear, every bone of it moves in a rigid manner. This is still the case if the trunk of an elephant is considered, where the highly non-linear nature of this object could still be *approximated* using an increased number of overlapping local linear transformations.

The previous observations describe a current challenge for computer vision. Is it possible to develop a method, and thus an algorithm, capable of automatically finding the appropriate deformation between two similar objects and infer at the same time the structure and the

complexity of such a deformation between objects?
The first evident fact in order to solve the above question is that the shape representation of the object is a crucial property to be taken into account. The reason is that shape communicates inter alia the geometrical information about the deformation of the object (s. section 1.4). However, to fully answer this question several challenges are required to be approached:

- **Shape Correspondence:** Given two different objects, it is crucial that the system establishes the correspondence between different parts of both shapes. The more both shapes differ from each other, the more difficult it gets to solve this problem. Furthermore, it is necessary to have a shape representation which allows local comparisons within the shape. For instance, this would not be possible if the system represented the object by means of a bounding box.

- **Modeling the differences:** An appropriate model for describing the transformations between shapes is required. For instance, highly non-linear mappings (e.g. Thin Plate Splines [249]) are able to transform a shape to any other arbitrary shape, but it is hard to discover the local structure of the shape using this model (s. chapter 7). On the other hand, piecewise models are flexible enough to describe a global non-linear transformation. However, the parts belonging to each component, as well as the complexity of the model (that is, the number of affine components required) in the transformation model still need to be inferred.

- **Lack of training samples:** The aim is to describe the transformation between objects and discover the structure of the object simultaneously, without prior knowledge of the class of the object or the shape. Therefore, it is not possible to learn a statistical shape model using training data.

- **Complexity of the model:** The model should be able to automatically adapt itself to the degree of linearity or non-linearity of the deformation between both shapes. For instance, in the case of a piecewise model, the algorithm should automatically find the number of transformations required to describe the shape change. Whereas the shape change of rigid objects will require a single transformation, a higher complexity is required for articulated objects.

- **Changes in viewpoint and scale:** The inference of the shape change between objects from a single pair of images presents an additional challenge. Already slight changes in viewpoints between images induce distortions in the underlying shapes. Thus, the overall transformation model requires partially coping with this fact. However, it is clear that in an extreme case where the viewpoint between images completely changes, the system will not be able to establish any correspondences between the shapes. Nevertheless, slight viewpoint changes still present a difficult challenge that needs to be handled.

- **Robustness against noise:** Commonly, the objects are not manually preprocessed to clearly *segmentate* the shape from the background. Therefore, the shape of an object needs to be automatically extracted in a robust manner and the registration process requires to robustly take clutter noise into account.

1.6 Objectives of the Present Thesis

The objective of this thesis is to study and devise new methods centered on the alleviation of the problems derived from the challenges described in section 1.5. At the heart of this thesis lies the conviction that shape is a crucial cue for solving these problems. Concretely, regarding the problem of simplified object representations, new models featuring a richer description of the object shape at different levels of the representational system (s. section 1.5.1) are developed. Furthermore, this thesis not only concentrates on search tasks but it also addresses the problem of modeling the understanding of shape similarity and shape changes as described in section 1.5.2. This analysis is a crucial step after the detection process in order for the system to understand its structure: Understanding a scene is not only about finding objects and their class membership within it, but it is also about understanding what the objects are and how they relate to each other.

Specifically, on the issue of object recognition the aim is firstly to devise a method for using curvature information about the object shape in the category model which is learned. The usage of curvature information is crucial since natural objects intrinsically exhibit curved boundaries and therefore, this information should be included in the detection system. Furthermore, since the importance of curvature for visual search tasks in human perception has been confirmed in different studies within the perception community (e.g. [262]), it seems advisable to integrate curvature cues in automatic object detection systems. Since a generic and stable representation of the curvature of objects is required to be used across different categories this task of devising a richer object representation is a non-trivial task to be solved.

Secondly, a new model for object detection is presented in order to go beyond bounding-box representations for objects. Visual recognition requires learning object models from training data. Commonly, training samples are annotated by marking only the bounding-box of objects, since this appears to be the best trade-off between labeling information and effectiveness. However, objects are typically not box-shaped. Thus, the usual parametrization of object hypotheses by only their location, scale and aspect ratio seems inappropriate since the box contains a significant amount of background clutter. Therefore, an approach is presented for learning object models for detection while simultaneously learning to segregate objects from clutter and extracting their overall shape. For this purpose, we exclusively use bounding-box annotated training data.

Finally, this thesis also investigates the capability of detecting shape equivalence (s. section 1.4) for object detection in unsupervised scenarios, that is where no annotation of training data is available for learning a statistical model. For this purpose, a dataset of historical Chinese cartoons without annotated data is analyzed with the goal of detecting important objects that may reveal within an iconographic analysis of the images important shifts in style or reveal the intention of the persons who commissioned the images. However, due to the lack of annotated training information, common object detection models cannot be learned and thus, a method is developed to indirectly infer the position, size and scale of important objects by detecting re-occurring shape patterns in the image, that is through the detection of *shape equivalent* objects.

Furthermore, since a visual system does not limit itself to detect, classify and characterize objects through global properties, the second part of this thesis focuses on the problem of describing changes between shapes. Specifically, a new model for analyzing the shape change between a single pair of images under missing correspondences is devised. The underlying idea is to describe the overall complex deformation of an object, using a piecewise linear model, which automatically estimates its complexity and at the same time is able to analyze the structure of the object by identifying each of the regions in the shape whose deformation can be described by a single affine component. This model is then used to analyze subtle modifications between an original artwork and a reproduction of it. The nature of these deformations is either due to deliberate alterations or due to geometric errors accumulated during the reproduction process of a certain image. For instance, an example of a deliberate alteration between a preparatory drawing and the finished artwork would be a conceptual change that induces alterations in the relative position of extremities in a human pose. Thus, in this case it is of interest for art historians to recognize the parts that feature the same transformation and determine to which extent these parts differ from other regions in the image. The second class of deformations is more subtle and is related to the drawing process itself. Copying images at that time in many cases was accomplished by placing a thin tracing paper on top of the original and sketching the contours. Movements of the semi-opaque sheet by the artist induced slight alterations in the reproduction. This art analysis represents a new interesting application within computer vision that is analyzed in this thesis for the first time. Finally, this method also shows how productive the interaction between computer vision and the field of the humanities can be in order not to supplant human expertise, but to enhance connoisseurship about a given problem.

1.7 Original Contributions

This section summarizes the contributions of this thesis

- The question of how to use shape to enrich object representations (s. section 1.5.1) is addressed. We present a novel view-based object detection model that efficiently represents an object shape, using both orientation and curvature features. It *directly* encodes curvature statistics and uses this shape cue together with orientation of gradients to perform object detection. The model exhibits competitive performance on standard databases.
- It is shown that curvature information can be easily integrated into all state-of-the-art representations that are based on gradient histograms with a low computational cost. Furthermore, this approach provides evidence that curvature cues provide complementary information that significantly enriches the widely used orientation histograms.
- The usage of shape for view-based object detection is further studied by presenting an approach that is capable of learning object models for automatic detection by explicitly representing object shape and segregating it from the background *without*, however, requiring manual segmentation of the training samples.
- A novel approach for learning a prototypical set of segments capable of representing all training objects of a given class is presented. Learning the object model based on this prototypical set of segments is then cast as a max-margin multiple instance learning problem. The learned model is therefore capable of detecting objects and assembling their overall shape simultaneously by grouping data-driven generated constituent shape segments of the corresponding object.
- The usage of shape equivalent objects (s. section 1.4) for unsupervised object detection is demonstrated. The position, scale and size of different objects is indirectly inferred in an unsupervised manner by detecting re-occurring shape patterns across a

large dataset of historical Chinese cartoons that were drawn during and immediately after the Chinese Cultural Revolution.

- Regarding the analysis of shape, a novel method for analyzing shape changes under missing correspondence between shapes is presented. At the same time that correspondence between point-set based shapes is found, the different shape constituent groups that are affine-transformed are inferred.
- Therefore, a piecewise affine registration model is conceived that is capable of automatically finding the shape groups that correspond to the different affine transformations. This problem, given the correspondences, is cast as an integer linear program that assigns points to transformations, based on their registration quality.
- The complexity of the piecewise affine model, that is, the optimal number of transformations used by the model is automatically found using a stability-based analysis. The shapes to be registered are randomly subsampled and the registration is carried out for different numbers of transformations. The most stable registration yields the corresponding number of transformations.
- A novel application for analyzing the shape changes between artworks is introduced. The usage of a piecewise affine model capable of automatically finding the parts of an art image that are transformed similarly enable the user to develop insights into which semantical parts were similarly reproduced and which were altered during the reproduction of an artwork.

1.8 Organization of the Thesis

This thesis is organized as follows:

Chapter 2 first gives an overview of the different components of a visual object recognition system and their general modeling paradigms. Furthermore, it also examines how shape information has been used to represent objects within the different modeling paradigms. Thus, we present a framework for classifying the underlying shape model of a given object detection system and at the same time we give an overview of state-of-the-art.

Chapter 3 develops a new model for automatic object recognition which uses curvature information for solving the task. Thus, we show that our enriched object representation improves the performance of a given detection system and by doing so, we address the question described in section 1.5.1.

Chapter 4 elaborates further on the alleviation of the problem described in 1.5.1 about the richness of object representations. This is done by introducing a novel method of learning object models for detection by explicitly representing object shape and segregating it from the background *without*, however, requiring manual segmentation of the training samples. The segregation from the background is not carried out as a post-processing step after having localized the bounding-box surrounding the object, but rather it is carried out during the detection process itself.

Chapter 5 exemplifies how the human visual capability of perceiving shape equivalence (s. section 1.4) can be used for unsupervised object detection. A method for analyzing a large dataset of Chinese cartoons drawn during the Chinese Cultural Revolution and thereafter is presented.

Chapter 6 is devoted to shape analysis and it gives an overview of the different choices made within the field of computer vision to model change between shapes.

Chapter 7 introduces a novel method for shape registration using a piecewise affine model which automatically estimates its complexity. Furthermore, the model automatically infers the parts in the shape which belong to the different affine components of the model.

Chapter 8 presents the conclusions of this present thesis.

CHAPTER 2

USING SHAPE FOR OBJECT RECOGNITION

The objective of this chapter is to give an overview of how shape has been modeled by object recognition systems. However, before one can understand the integration of shape it is essential to first understand the general modeling paradigms lying at the heart of each system. As stated in section 1.5.1 different description levels in an object recognition system can be distinguished (s. figure 2.1): (a) the feature representation that captures the low level content of the image. (b) the object model representation and (c) the final representation of an object instance, which can be used then for a further understanding of the image content. Using this structure as an orientation we will describe different modeling decisions and state-of-the-art in sections 2.1 and 2.2. Thereafter the integration of shape will be discussed in section 2.3.

2.1 Local Image Descriptors

The basis for many high-level tasks in computer vision and specifically for object detection systems consists of the representation of specific structure in the image data. This representation is commonly referred to as the *local image descriptor*. At the basis of each descriptor lies a crucial modeling decision which is inevitably confronted with a trade-off between the information content and the processing cost of this information. In some cases, highly detailed descriptions may help to solve the task at hand, but this comes at the cost of dealing with more data and processing resources. Furthermore, the importance of choosing an appropriate low-level description of the image data is given by the fact that all information that is lost at this stage cannot be recovered since all further steps in the system are dependent on this representation. Finally, a local image descriptor can be classified depending on which information is encoded in the image and on how it is done. Whereas local image descriptors may use statistics about orientation of gradients ([156, 158, 178, 29, 60, 17]), other descriptors only use pixel intensity values ([11, 47, 191]), or geometrical relations between discretely sampled landmark points ([13, 68]) for their computation.

In the following we will briefly review some of the most popular local image descriptors in the field, and observe that depending on the kind of information that is encoded, shape information may or may not be used. For instance, whereas the usage of orientation of gradients can be considered as a *shape-based descriptor* since these descriptors provide a local representation of the geometry of the object, color histograms or simple intensity-based descriptors evidently only encode local *appearance information.*

SIFT Features

The scale invariant feature transform (SIFT) [156, 158] in its original version, is a histogram-based representation of gradients in a local patch. Previous steps to the local image descriptor consisted of assigning a dominant scale and orientation to the position where the SIFT feature is being computed. The whole region is then rotated according to the dominant orientation in order to achieve invariance with respect to rotations. Thereafter, the region is subdivided into a regular 4×4 grid and 8 bin histograms of the gradients orientations are calculated resulting in a 128 dimensional vector. An interesting fact is that because the SIFT descriptor is normalized, the gradient magnitude changes have no effect on the final descriptor.

A variation of the SIFT descriptor was presented in [178]. This variant called GLOH considers larger spatial regions for the histograms and the dimensionality of the descriptor is further reduced to 64 dimensions through principal component analysis (PCA). However, not only gradient information of gray-scale images has been used in the literature to calculate SIFT descriptors, but also color information has been encoded in different ways. For instance, whereas Bosch et. al. [29] computed SIFT descriptors over all three channels of the HSV color model, Van de Weijer et al. [244] introduced a concatenation of a saturation-weighted hue histogram with the SIFT descriptor. Furthermore, in [242] the performance of descriptors that combine color information and the traditional SIFT descriptor has been analyzed. Although the method for computing color-SIFT and the traditional SIFT descriptor are similar, it is important to remark that whereas color-SIFT does not encode any local shape information about the geometry of the object, the traditional SIFT descriptor can be considered as a shape-based descriptor.

SURF Features

This image descriptor was first introduced in [11] and it is strongly inspired by SIFT. This feature also calculates a dominant orientation and scale for each region, then the region is split-up in regular 4×4 square sub-regions. However, instead of calculating histograms over orientations of gradients for each sub-region, Haar wavelet responses (in horizontal and vertical direction with respect to the dominant orientation) are calculated. The wavelet responses and the magnitudes are then in turn summed up over each region to form the entries of the feature vector.

Shape Context and Geometric Blur

Shape context [13] is a feature specifically designed to describe object shapes represented by landmark points $\{p_i\}_{i=1}^n$. The basic idea is to consider for each point p_i the $n-1$ chord vectors which connect all other points in the shape to p_i and build a histogram (in

the log-polar space) over these vectors. This histogram is considered as the shape context of the point p_i. Similar to this feature, however, without building a histogram, in [68] relative orientations between specifically chosen chords for each point are used, resulting in a matrix instead of a vector descriptor.
The geometric blur descriptor [18] can be considered as the continuous version of shape context. Around an interest location the region is blurred with a spatially varying kernel. Gaussian kernels with a standard deviation proportional to the distance from the center of the region were used in [17, 16]. The objective is to put emphasis on the center of the region and gradually suppress the importance of regions lying further away. Normally, edge signals are filtered resulting in the high dimensional geometric blur descriptor.

Histogram of Oriented Gradients (HoG)

N. Dalal and B. Triggs presented the histogram of oriented gradients (HoG) in [60] with an application to pedestrian detection. The idea behind this very successful and widely used descriptor is to depict the local shape appearance of the object by means of the distribution of weighted oriented gradients. The region of interest is first subdivided into small connected regions (called cells), and for each cell a histogram of gradient directions is calculated. The concatenation of these cells builds the final descriptor. In its original version, 9 bin histograms over $0 - 180$ degree orientations were used for the histograms in each cell. Each pixel contribution was weighted with the gradient magnitude itself. In order to account for changes in illumination and contrast the gradient strengths were locally normalized which required grouping the cells together into spatially connected blocks. N. Dalal and B. Triggs used 3×3 cell blocks of 6×6 pixel cells with 9 histogram channels. However, a variation of the original HoG feature was introduced by P. Felzenszwalb [80] and has recently become very popular ([273, 77, 9, 99]) for object recognition. In this variant, the calculation of the histograms for each cell is changed. In praxis the authors found that for some object categories, recognition performance increases using contrast sensitive features (B_1), while some categories benefit from contrast insensitive features (B_2). Whereas for B_1, a weighted histogram (using the gradient magnitude) with a 9 bin discretization of the gradient orientation is used, B_2 is calculated using 18 bins. In addition to both features a 4-dimensional vector capturing the overall gradient energy in square blocks of four cells around each pixel (i, j) is added, resulting in a $9 + 18 + 4 = 31$ dimensional vector for each cell. The overall HoG feature consists then in the concatenation of the vectors for all cells over the region of interest.

Binary Robust Independent Elementary Features

This appearance based local image descriptor called BRIEF [47] is constructed by first smoothing a square area of interest with a Gaussian Kernel. Hereafter, intensity values $p(x)$ and $p(y)$ between n_d randomly sampled pairs of locations (x, y) within the area of interest are compared (called tests) and the BRIEF descriptor is built using a $128, 256$ or 512 dimensional binary string containing 1 at the i-th position if the i-th comparison yields $p(x) < p(y)$ and 0 otherwise. The tests used are sampled from an isotropic Gaussian distribution with fixed standard deviation. Rublee et al. [213] presented a modification of BRIEF, which is rotation normalized by first rotating the patch according to a previously calculated predominant orientation.

Local binary patterns

This appearance-based descriptor is better known as LBP and was first described in 1994 [191]. Since then, it has been used mainly for texture classification and has shown a good performance in combination with HoG descriptors [254]. In its simplest form, the window of interest is divided into cells (e.g. 16 × 16). Thereafter, each pixel in a cell is compared with its eight neighbors (clockwise or counter-clockwise). An eight digit binary number is built using the pixel's value compared with its neighbors, resulting in 1 if the pixel's value is greater than its neighbor or 0 otherwise. Afterwards a histogram is built for each cell, the feature vector results out of the concatenation of all these histograms.

2.2 General-Modeling Paradigms

Once that image-data is encoded into discrete features, the next stage in the representational scheme of an object recognition system consists of a model capable of binding the different features into a joint representation of the object which enables the system to make a decision about the presence or absence of a certain object belonging to a certain class. Many models have been developed throughout the history of computer vision for this purpose. Even though these models substantially differ from each other, they also share common concepts or thought patterns which can be classified into a general scheme or framework capable of providing an overview about the field of object recognition. For instance, [193] described such a general modeling scheme utilizing four different attributes, which directly refer to the mathematical model behind the object representation. In the following, such a scheme is used to recapitulate and systematize current state-of-the-art approaches for recognition.

Model-based vs View-based Models

Model-based vision systems look at the world from a geometrical viewpoint. For these systems describing an object consists of inferring geometrical constraints from different views of the object in order to obtain a geometrical 3D model. Therefore, object recognition consists of the process of fulfilling geometric constraints between the current scene and the object model in order to make a decision about the presence or absence of the object. The origins of model-based vision can be traced back to Lawrence G. Roberts and his seminal work [212] where the foundations, formulations, and goals for model-based vision were established. In a further step [251] formulated the problem of vision as a problem of satisfying constraints, where projections of 3D objects could be labeled according to their 3D configuration from a single 2D view. Furthermore, D. Lowe in his paper [157] introduced the concept of *hypothesize and test*, where each algorithm is able to hypothesize a correspondence between a collection of image features and a collection of learned object features, and then uses this correspondence to generate a hypothesis about the projection of the object model (called back projection). In a final step the algorithm compares the rendering to the image and in cases of sufficient similarity, the hypothesis is accepted [94]. The advantage of a model-based viewpoint consists of the fact that geometrical modeling is invariant to factors like luminance, or viewpoint changes, however, in praxis the reliable extraction and hypothesizing of abstract geometric representations is difficult.

In contrast to model-based methods, during the 90's *view-based* approaches emerged and have become very popular since they attain a very good performance in praxis (e.g.

[61, 165, 80, 61, 140, 103]). View-based approaches aim at learning a model of the object's appearance in a two-dimensional image under different poses and illumination conditions. By this, they avoid constructing a 3D model of an object as well as having to make 3D inferences from 2D features. For instance, an early view-based approach is Poggio and Edelman [202] which demonstrated how a 3D object can be recognized using the raw intensity values of 2D images. Another view-based approach is the eigenfaces technique of Turk and Pentland [237], where a face image is represented as a vector of pixel values and the eigenfaces are the eigenvectors associated with the largest eigenvalues of the covariance matrix of the sample vectors. During testing, a query intensity vector is projected into the lower dimensional eigenspace obtaining a vector of weights describing the contribution of each of the eigenfaces. Thereafter, to determine the class identity of the query vector, the Euclidean distance to each class (represented by a vector) is minimized.

Holistic vs Part-based models

The term *holism* refers to the idea that systems should be viewed as a whole, not as a collection of parts. For instance, view-base models using *template matching* for object recognition (e.g. [203]), follow this holistic idea. In its simplest variant the shape of the object is captured by a binary template and the object is matched by translating and positioning the template at various locations of the image. This last procedure is called the *sliding windows technique* and is used e.g. in [219, 248, 81]. At each position in the image the distance of the pixel values of the image which lie under the data pixels of the template is estimated. Based on the matching score, a decision about the presence or absence of an object instance can be made. Furthermore, invariance to scaling or rotation can be achieved by additionally searching with scaled and rotated versions of the template. However, the template is not restricted to be a binary numeric mask, but it allows more complex templates like [60], where a vector-based template is learned from the training data and is slid across the image, using convolution operations. Another line of research within the holistic paradigm is the contour-based method like the work on snakes by [127]. Kaas et al. understands under the *snake* a spline which is fitted to lines and edges based on an energy-minimization procedure trades prior knowledge about contour against evidence from an image.

On the other end of the spectrum, *part-based* models [93, 136] introduce spatial structure in the object representation in order to alleviate the limitations of holistic models when handling highly deformable objects (e.g. articulated objects). Fischler and Elschlager [93] proposed for the first time a model that combines local templates arranged according to a geometric configuration. In this case, recognition works by solving the correspondence between local parts in the image and the model while the global distortion is minimized. Among part-based models, the geometric configuration between local parts can significantly vary, ranging from independent parts like in *Bag-of-Words* models [56, 224, 84, 243], to fully connected parts as in *constellation models* [34, 255, 256, 86, 87]. In the middle between both extreme systems modeling tree-structured geometrical relations between parts (first proposed by Felzenszwalb and Huttenlocher [82, 83]) can be found. These tree-structured models can be seen as predecessors of the work introduced in [80] by the same author. This work has been awarded with the PASCAL VOC "Lifetime Achievement" prize in 2010, for its influence in the computer vision community. The model consists of a tree-structured part-based model, where each of the parts is described by a HoG feature and only relations to the root-node of the tree are modeled. Objects are found using sliding windows over the image at different scales, and at each position a score of the model is evaluated. This score is made out of the matching score of the root-template plus the sum over the parts matching scores on their location minus a deformation cost measuring the deviation of the part from its ideal location (relative to the root) learned during training. The templates are learned based on HoG features using a variation of the linear Support Vector Machine ([69]) capable of jointly learning the templates and the optimal position of the parts with respect to the filter.

Generative vs Discriminative Models

The distinction between generative and discriminative models refers to the paradigm how a model is learned from a set of training images. *Generative models* pursue the learning of the optimal class label y of an object $\mathbf{x}$ by learning the joint probability density function $p(\mathbf{x}, y)$ (or equivalently $p(\mathbf{x}|y)$ and $p(y)$ since $p(x, y) = p(x|y)p(y)$). Classification is then performed via the Bayes' formula

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}|y)p(y)}{p(\mathbf{x})} \tag{2.1}$$

$$\propto p(\mathbf{x}|y)p(y) \tag{2.2}$$

Examples of popular generative approaches are [85, 147, 218]. Among the benefits of this class of methods, three advantages can be enlisted [238]: (a) these models can handle missing data or partially labeled data, (b) a new class y_2 can be added incrementally by learning the class conditional density $p(\mathbf{x}|y_2)$ independent of the other classes, and (c) generative models can handle object combinations (e.g. faces with glasses), which were not seen during training. However, if the goal of a detection system is interpreted as a decision whether a certain object hypothesis belongs to a certain class or not, learning a class in a generative manner seems a far more complex task than only learning the decision boundary that separates instances of a class from other classes. And in fact, V. Vapnik [51, 69] formulated this objection when he introduced the Support Vector Machines (SVM), a *discriminative* classifier which has become a standard classifier for discriminative object models (e.g. [80, 61, 165]).

Discriminative approaches for object recognition generally work by directly learning a parametric model for the posterior probabilities $p(y|\mathbf{x})$ from a set of labeled training data. This class of approaches can be understood as learning a direct mapping from the object representation to the label space, that is, the mapping $y = f(\mathbf{x})$. For instance, *Support Vector Machines* intend to find a hyperplane, which achieves a maximal separation between two classes so that the distance from it to the nearest data point on each side of the hyperplane is maximized (s. Sec. 3.5.1 for more details). Different advantages using discriminative methods can be observed [238]. For instance, discriminative methods are typically very fast in the prediction of a new sample. Furthermore, whereas generative models may model details of the distribution of $\mathbf{x}$ in irrelevant regions for determining the posterior probabilities, discriminative approaches use their flexibility in regions where the posterior significantly differs. Finally, there exist indications that in the asymptotic case of large training sets, discriminative approaches yield lower error rates than generative ones [189].

Hierarchical vs Shallow Models

Hierarchical models were originally used by biologically inspired object recognition systems. For instance, an early system is the *Neocognitron* [97], a multilayered network consisting of a cascade of alternating layers of neuron-like cells where the C-cells and the S-cells are the most important ones. Whereas S-cells are only activated when particular features are present at a certain position in the input layer (e.g. lines with certain orientation, corners, end points etc.), C-cells receive signals from groups of S-cells which extract the same feature at different positions. The model starts with a 2-D array of pixels as input and presents the classification result by the activation of cells in the output layer. Starting with the Neocognitron many hierarchical models relying on *convolutional neural networks* have been introduced in the literature, the HMAX system [211] and the LeNet-5 system [142] are prominent examples of this. The idea behind a convolutional neural network consists of learning a hierarchy of features where higher-level concepts are defined from lower-level ones [15]. In recent time the construction of such models has been the goal of a field within machine learning called *deep learning* that pursues the usage of deep hierarchical structures for solving different tasks in Artificial Intelligence. For instance, a convolutional neural network for classification in the presence of large amounts of training and test data was recently introduced in [135]. This network consists of 60 million parameters and $650,000$ neurons distributed across 5 layers. This achievement has partially been made possible due to very efficient parallel GPU-based implementations of convolution operations.

In contrast to hierarchical models view-based models have typically concentrated on shallow models which perform classification as directly as possible in the image space. However, not only non-hierarchical, view-based models like template matching [203] or simple bag-of-features [57] approaches exist, but also hierarchical view-based models like [273, 80] can feature a flat hierarchy which is normally used to introduce spatial structure between the number of model parts. However, the deeper the structure, the higher the complexity that these models require to deal with. For instance, the recent model of [273] obtains 25947 dimensional features arising from a three-layered structure of HoG features. It becomes evident that training a discriminative model in this highly-dimensional space requires very large amounts of training data in order to overcome the *course of dimensionality.* This phenomenon first formulated by Richard E. Bellman in [12] refers to the fact that when the dimensionality of the data increases, the volume of the space increases and the available data becomes sparse. This sparsity is problematic when a distribution from a finite amount of data is learned, since sufficient samples are not guaranteed.

2.3 Shape-Modeling Paradigms

The modeling framework presented in the last section is very useful to describe the general structure of a given model for object recognition. However, the major shortcoming consists in the limited ability of giving specific modeling insights about the codification of the object's shape itself. For instance, if we consider a part-based model this scheme will not be able to distinguish if the underlying shape model operates with edge contours or if it uses segments or regions for describing the overall form of the object. Furthermore, that framework will not take into account the degree of supervision required for constructing the underlying shape model. Therefore, in the current section we will extend the framework of section 2.2 with the aim of understanding different paradigms and aspects of how shape has

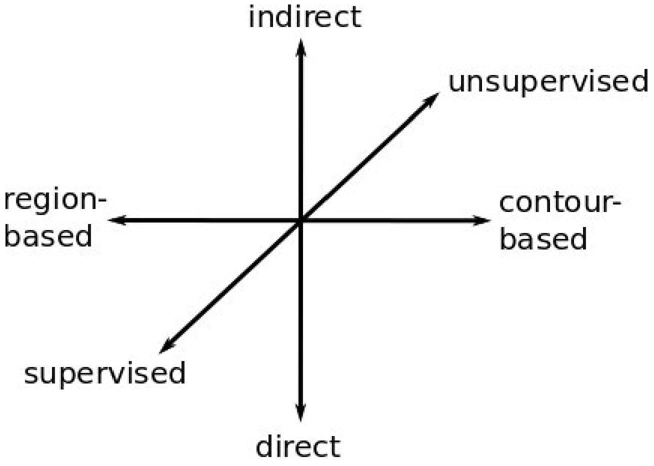


Figure 2.1: Different paradigms for *shape modeling* in object recognition systems. For more details see section 2.3

been integrated into state-of-the-art object recognition systems. This framework consists of three attributes which will be described in the following section.

2.3.1 Indirect vs Direct Shape Representation

Indirect shape representations[1] are best explained by *Bag-of-Features* models [56, 224, 84, 243] using shape-based local descriptors (e.g. SIFT, see section 2.1 for more details). In these models the object is represented by a distribution over a codebook of characteristic features. During the training phase, features are collected from the training data and these features are clustered to obtain a codebook. Thereafter, a classifier is trained using the bag-of-features. For a better localization of the object, sliding windows together with a prune technique can also be used. From this it is clear that in a bag-of-words approach every spatial arrangement of the features is disregarded and only the co-occurrence is captured by this model. The only information about the geometry of the object is *indirectly* captured by the features used for the construction of the bags (if the underlying local descriptor is a shape-based descriptor). Thus, indirect shape representations refer to models, where shape cues are only captured at the level of the featural representation (that is, the first level of the representational system as described in section 1.5.1) and not at the model-level itself. Although an extension of the bag-of-features approach was presented by Lazebnik et al. [140] who introduced a coarse spatial information into the bag-of-features by subdividing the image into a regular grid and building separate feature bag descriptors for each cell individually, to consider this grid-like structure as the true shape of the object is not straightforward. Moreover, in cases using shape-based descriptors for building the different bags this model also encodes local geometric information about the shape of the object also in an indirect and local manner. Lazebnik's model has a certain similarity to rigid template matching methods (e.g. [60]), where the shape information about the object is also indirectly given by the local image description based on histograms of oriented

[1] This terminology should not be confused with the Implicit Shape Model approach introduced by [144] which is based on a voting approach and thus, using our terminology can be considered as a direct shape representation

gradients.

Opposed to an indirect representation we find a vast amount of models where the shape of an object is *directly* used in the model itself. An example of this class of models is the constellation model. Perona et al. [34, 255, 256] introduced this model, where the joint configuration of all local parts (encoded through local feature descriptors) is modeled. The dependencies between the parts can be thought of as a fully connected graph, where a single part depends on the rest of the parts, this means that no further conditional independence is assumed. Another class of models which shares the idea of *directly* modeling the shape of the object are hough-voting-based approaches [145, 143, 91, 165, 98, 266, 194]. The idea behind these methods consists in letting local features vote for object hypothesis (parametrized by location, scale and aspect ratio). These methods directly model the shape of the object insofar as every local feature (or collection of features) votes for an object center hypothesis, and once a location is accepted as the location of a class-object, the shape of the object can be recovered by selecting the features that voted for this concrete location. In approaches like [228] or [267] this idea of directly recovering the shape can be better appreciated. For instance, [267] learns a contour shape model within the Multiple Instance Learning framework where multi-instance sets of contours in an image are considered. Using these bags the discriminative model recovers the entire object shape and by doing this a decision about the class membership is taken. Finally, region-based approaches as described in the next section, are also considered by the current framework as direct shape representations.

2.3.2 Contour-based vs Region-based Shape Representation

An important decision in the modeling of shape is whether direct contours or rather regions (e.g. using a combination of superpixels as in [106]) are used to describe the object's shape. Although contour-based methods are vast, several lines of work can be distinguished. Besides contour-based voting approaches like [120, 266], active shape models [54, 215, 92], contour-based shape hierarchies [92] and partial shape matching [210, 228, 161] approaches can be found. Also template-matching techniques using contours like [120, 154] belong to this class. The idea behind [195, 120] consists of using a weighted contour-based version of the hough-voting paradigm. During training, a codebook of contours is gathered and their relative location is kept. The relative importance of each contour is learned using Adaboost. Thereafter, during testing, the codebook contours are matched to the edge map of the test image and hypotheses are generated. Furthermore, approaches like [210, 228, 161] rely on partial matching of the training edge fragments against fragments of the query image to perform object detection. In some cases (e.g. [210]) hand-drawn models for each category are required. However, [228] treats long training contours as latent variables and their placement for the shape model is learned using a latent SVM.

Methods using shock-graphs for describing the shape of the object are also contour-based [223, 162, 10]. These methods rely on an exact characterization of the silhouette of the shape given by a shock-graph, which is a labeling and partitioning of the skeleton points (shocks) making up the medial axis transformation of a shape. This shape description relies on an excellent segmentation of the image which yields the unoccluded contours of the object's shape. For this reason many of these methods have been typically applied only to silhouette-based recognition where the shapes contain unocluded presegmented closed contours. Contour-based shape representation models rest on the foundation of a reliable extraction of contours in an image, which is a difficult task in real-world images.

Complementary to contour-based approaches, region-based models have also been popular

for object recognition. An important class of these models are subsumed under the term *segmentation-based object recognition* [99, 172, 247, 37, 106]. In [37], the authors first generate a set of class-independent pixelwise figure-ground segmentation masks, distributed across the entire image using the bottom-up method [38]. These masks are extracted automatically without prior knowledge about the corresponding class, by solving a sequence of constrained parametric min-cuts problems on a regular image grid. Thereafter, the segments are ranked based on their plausibility of being an object. This is achieved by training a classifier using ground-truth segmentations provided by humans on the Berkeley Segmentation Dataset. Once the figure-ground segmentation masks for all images are provided, the authors in [**?**] assume that the best ranked segment within the bounding-box of each positive example covers the entire object. This segment is thus used in turn to learn a regression function that predicts the quality of query segments for being an object of the desired class. In a similar fashion, [106] proposed a method for detection using regions. The authors construct a tree using the hierarchical segmentation engine [3], where each segment represents a node of a tree. Each segment is then represented by different local features calculated on a regular grid that is superimposed on every segment. During training, discriminative weights are learned to estimate the importance of each segment. Finally, during testing, a hough-voting scheme based on segments is used to vote for different object centers, which estimate the location and scale of the objects.
Not only segmentation-based systems belong to the class of *region-based shape models* but also models like deformable template matching algorithms belong to this class of methods, since the different templates can be viewed as constituents of the shape. Deformable template matching methods [268, 123, 80] were introduced to compensate the limitations of template-matching methods in the presence of articulated objects. These models apply a global transformation during the matching process and the objective is to minimize the deviation between the model template and the query image. In practice bounded deformations are normally preferred during the energy minimization.

2.3.3 Supervision Degree

Depending on the amount of information available during training, different shape models can be learned. In addition to bounding-box annotations for the positive samples *supervised methods* require also manually-annotated information to learn the model. For instance, [82, 83] proposed a part-based model for recognizing people in images where the spatial relation between parts are tree-structured. Training requires manually labeling the part configuration in training images (thus only a small number of parts are used). Another supervised part-based model with manually-labeled parts is the original version of the k-fan model [53]. These k-fans are graphs with dependencies between parts, where the parameter k determines the number of parts considered as reference parts, and all other parts are dependent on the reference parts. Furthermore, methods like [99, 172, 247] are region-based models with strong supervision since manually labeled figure-ground segmentation masks are required during training. For instance, [99] augments the bounding boxes with a set of binary variables, each of which corresponds to a cell of the HoG feature representing the object. The model is learned using the structured output framework [235] and is used to improve detection results obtaining a richer output of the detected object. The main disadvantage of this class of methods is that manually-labeled information is usually not available for large-scale detection tasks or it is tedious and expensive to obtain. Recently, the usage of Amazon Mechanical Turk for gathering this additional information has become popular (s. [64] and references therein). Mechanical Turk is a crowdsourcing

Internet application that enables the coordination of human resources to perform tasks that computers cannot do. By these means large amounts of richly-annotated data has become available for training models. However, the usage of such a system for research in computer vision presents several difficulties which should not be ignored. Firstly, the amount of annotated information is proportional to the amount of money invested and thus an imbalance in research due to the larger monetary power is introduced. Secondly, it is difficult to evaluate whether the people annotating large amounts of data are doing this freely or whether they require money as job income. In the latter case it is worth asking if the money spent is a fair remuneration for the completed work. Finally, using a fine-grained object annotation may lead to a fine-grained object detection with high performance [64]. However, the more challenging question of obtaining a fine-grained object detection using a weak or even using any supervision at all remains unanswered. It should not be forgotten that every annotation used for training a model is an implicit concession to the fact that computers are still not able to automatically infer this information directly from the image itself.

On the other end of the spectrum we find shape models which are learned using only weak supervision (e.g. using only manually annotated bounding-boxes around the objects of interest) or without any supervised information at all (e.g. [255]). Examples of weakly supervised methods are [90, 267, 228]. For instance, the contour-based model [90] constructs a shape model by finding the contours which consistently reoccur within the bounding-boxes across training instances at similar locations and scales. The relevant segments are then found maintaining separate voting spaces for different segment types. The local maximum over these spaces finally yields a model-part having a specific location and size relative to the training bounding-box. Finally, [255] probably has been the first work to tackle the problem of *fully unsupervised* object detection by means of a direct shape model. The authors build their method on a simplified constellation-like model. First, they extract highly textured regions in the training images. Using feature selection, they select only a subset of these regions to learn a generative model by means of the expectation maximization (EM) algorithm. Object detection is then performed by localizing parts and building object candidates. Thereafter, the learned probability density is used for calculating the likelihood that a given hypothesis arises from an object.

CHAPTER 3

BEYOND STRAIGHT LINES - OBJECT DETECTION USING CURVATURE

3.1 Using Curvature for Object Recognition

As stated in section 1.4, the representation of shape is one of the most fundamental problems in the study of the human visual systems ([46]). Evidence has been gathered that the visual system is predisposed to detect a given image feature, event, or configuration [261]. In his review J. Wolfe ([262]) agrees with the consensus that there are about eight to ten basic features that play an important role for visual search tasks[1]: color, orientation, motion, size, curvature, depth, vernier offset, gloss and, perhaps, intersection and spatial position/phase. Regarding curvature as being a basic feature this finding is consistent with other works (e.g. Treisman and Gormican ([234], Foster and Westland [95])), which found that curved lines could be found in parallel among straight distractors (see also [31]). Moreover, when the target is straight and the distractors are curved, the search is less efficient. This suggests that curvature is a property whose presence is easier to detect than its absence and could be considered as a basic feature.

The psychologist Jeremy Wolfe [260] developed the *guided search model* in order to explain how preattentive processes are used to direct attention. A preattentive process consists of an accumulation of different signals or stimuli from the environment that are processed relatively quickly and unconsciously by the visual field building a saliency map related to the current individual thinking. According to Wolfe's Model the brain generates an attentional priority based on this saliency map. During a visual search task attention is directed to the item with the highest priority and if it is rejected by a conscious (or attentive) process the attention will move then to successive items. Many recent automatic object recognition systems in computer vision share similarities to the Guided Search Model of Wolfe (e.g. the deformable part model of [80] builds a dense map of feature responses which are then

[1] In this framework, a visual search task is an experiment where subjects are asked to look for an item among distractor items. Whereas, on some trials, a target is present, on the rest of the trials, only distractors are shown. Furthermore, the subject is instructed to give certain response to indicate that the target object was found and a different signal to indicate the absence of the target [262].

used to prioritize the object search) and the attempt to integrate different stimuli (e.g. in [245, 77] the usage of different types of features are analyzed) in order to search for objects of a certain class. The usage of features like color and orientation (s. section 2.1), have also reached maturity within the field leading to powerful detectors (e.g. [61, 165, 80]), while others like curvature have not received the same level of attention although some basic steps have been taken (e.g. [179]).

To yield robust powerful object representations the vision community has now broadly adopted the theme of histograms of gradients at the lowest level of the representational system (s. 1.5.1): Almost all present approaches, ranging from semi-local descriptors such as SIFT [156] to holistic object representations [61, 140, 103], are based on histograms of local gradient orientation. In effect, the usage of this representation results in a straight line approximation of object boundaries since local regions are described by a histogram over a discrete set of edge orientations that they contain. In this framework a smooth curve cannot be distinguished from one with sharp bends or from a set of differently oriented lines in arbitrary configuration as can be seen in Fig. 3.1. Moreover, natural objects actually do not exist in a blocks-world domain [225] and have not been designed with a ruler on a drawing table. Instead they do exhibit characteristically curved boundaries, e.g., consider the differences between apples and pears.

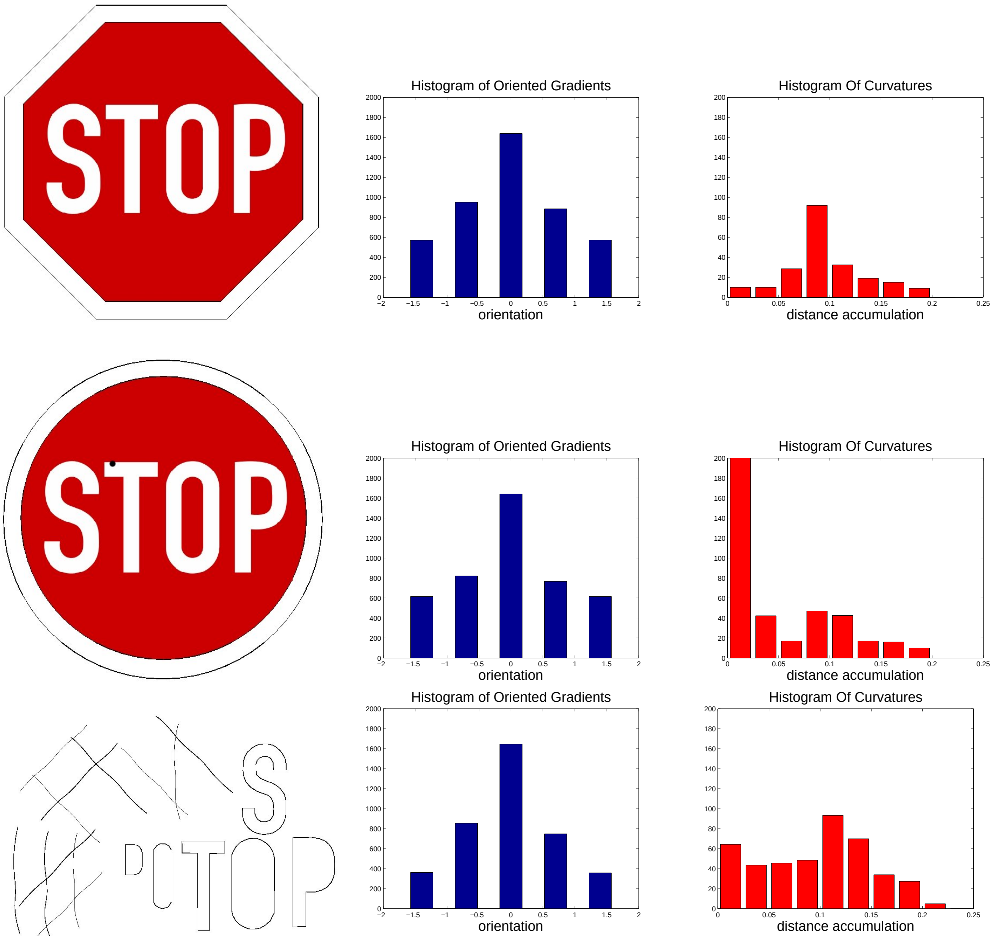
For these reasons, in the present chapter we extend the widely used object representation based on gradient orientation histograms by incorporating a robust description of curvature and show that integrating curvature information substantially improves detection results over descriptors that solely rely upon histograms of orientated gradients (HoG). The proposed approach is generic in that it can be easily integrated into state-of-the-art object detection systems. Furthermore, the present method directly deals with the problem in computer vision systems described in section 1.5.1. That is, since object recognition systems are based upon an abstraction process the main weakness of current methods resides in the fact that at each level of the abstraction process crucial information about the object is totally missed and is not used for later tasks to solve the final problem. For instance, if only orientation of gradients is used, the description of the object's shape will be very simple and rather crude. For this reason, the aim of this chapter is to develop a method which is able to solve the recognition tasks based on a richer and thus more accurate description of the object's shape leading to a higher performance of the system.

3.2 The Contribution

The present chapter describes a novel object detection system that efficiently represents an object shape using both orientation and curvature features. It *directly* encodes curvature statistics and uses this shape cue together with the orientation of gradients to perform object detection. The results presented in this chapter were published in [182] and have been further used by [77].

The insights gained by this method are threefold:

1. curvature information can be integrated effortlessly into all state-of-the-art object representations that are based on gradient histograms.
2. this representation has low computational cost
3. it provides complementary object information that significantly enriches the widely used orientation histograms.



(a) (b) (c)

Figure 3.1: (a) Original images, (b) Histograms of oriented gradients, (c) Histograms of Curvature. A smooth curve cannot be distinguished from one with corners or from a set of differently oriented lines in an arbitrary configuration based only on histograms of oriented gradients.

3.3 Curvature Estimation Methods

Curvature estimation in digital spaces, i.e. curves extracted from images, has been studied in depth and several methods have been proposed. [263, 108] estimate curvature as the derivative of the tangent direction with respect to the arc-length. Another way of estimating the curvature is used in [45, 240, 204] by calculating the osculating circle touching the curve. Curvature estimation methods based on the first and second derivative of the curve can be found in [14, 79, 153], where [79] estimates the derivatives in the frequency domain of a closed curve by a multi-scale convolution of the curve with different Gaussian kernels and [153] approximates the curve with a rotated parabola. Finally, [109] proposed an efficient new approach to approximate discrete curvature at a given point p by means of the accumulation of Euclidean distances from different secant lines to the point p. This method has proved to be more stable compared to the curvature-space method [179], where a boundary is represented as a parametric function of arc length, and inflection points are detected as stable zero-crossing points over convolution of the shape with Gaussian filters

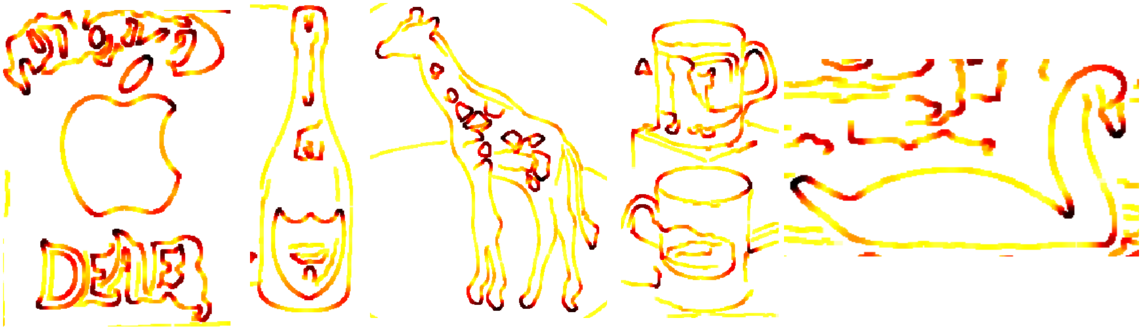


Figure 3.2: Examples of local curvature approximation used by our descriptor on the ETHZ Shape Dataset.

at different σ levels. Moreover, the calculation of curvature using [109] is extremely fast, thus making this approach ideal to be used for object recognition purposes.

3.3.1 Curvature Cues for Object Detection

Many methods using curvature information for finding interest points (e.g. high-curvature points) have been proposed in the literature [96, 5, 176] or more recently [111], [6]. However, the direct use of curvature information for building object descriptors has seen comparably little progress. The early approach of [179] works for object recognition under the assumption of closed boundaries. Furthermore, modern descriptors like k-AS [88] explicitly decide not to take curvature into account: "The proposed descriptor considers the segments as completely straight segments so as to capture only the relevant information of the geometric configuration they form, and not the unreliable details of the weak curvature along them" ([88], p. 9). Moreover, the simple, yet powerful descriptor proposed by Dalal and Triggs [61] used for pedestrian detection with further extensions in [165] and [80] solely encodes orientation of gradients in the form of histograms. Therefore, the aim of this work is to directly use curvature statistics in a discriminative way to improve object recognition.

Finally, related to our work is also the paradigm of sliding windows for object detection. Some work [257, 137] has been recently been devoted to alleviate the immanent efficiency problems (mainly computational cost) that this framework presents during object localization. [137] presents a branch-and-bound scheme to efficiently maximize certain classes of classifier functions. Very recently [257] proposed an efficient method for histogram computation and evaluation of classification functions that has a constant complexity in the histogram dimensions. This promises that the sliding window framework will remain a powerful tool for object recognition, especially in combination with histogram-based descriptors.

3.4 Robust Representation of Curvature

In this section we describe a method to perform object detection based on curvature information from shapes and use this information directly as a discriminative feature together with histograms of oriented gradients (HoG) [61]. We abbreviate the joint descriptor with **HoGC**.

A very fast and stable way to approximate the curvature for planar boundaries is to use the chord-to-point distance accumulation (or distance accumulation) [109]. Let B be a set of N consecutive boundary points, $B := \{p_0, p_1, p_2, \cdots, p_{N-1}\}$. The set of points is obtained by following the edge contours of objects in a clockwise direction. Each pair of points p_i and p_{i+l} defines a line L_i, where $i + l$ is taken modulo N. L_i depends on the parameter l whose adjustment is explained later in this section. For each point p_k the perpendicular distance D_{ik} from the line L_i is computed, using the Euclidean distance. The distance accumulation for a point p_k and a chord length l is the sum

$$h_l(k) = \sum_{i=k-l}^{k} D_{ik}. \tag{3.1}$$

[109] showed that equation (3.1) is more stable, regardless of different values of l, than in Gaussian smoothing curvature calculation methods, which give dislocation, broadening and flattening of the features ([259]). Furthermore, it was shown that in the analytical case, the chord-to-point distance accumulation asymptotically approximates (up to a constant) the true curvature of the boundary.

Given an image, we first extract edges using the Berkeley edge detector [173]. Connected components on the binarized edge map yield a set of segments B_j. Using these segments we calculate the distance accumulation given in equation (3.1). To be robust against the choice of l we choose a bank of values $\{l_1, \cdots, l_n\}$ ranging from between 5 and 40 pixels and take for every point p_i on segment B_j the median

$$c_j(p_i) := \text{median} \left\{ \frac{h_{l_s}(i)}{l_s^3} \;\middle|\; s = 1, \cdots, n \right\} \tag{3.2}$$

as a boundary feature. In Fig. 3.2 we show some examples of the curvature of natural images.

The idea behind the HoG descriptor of Dalal and Triggs [61] is that local statistics about intensity and orientation of gradients can encode the appearance and shape of objects. Curvature information of shapes can be encoded in a similar way. We divide the image into connected cells and for each cell we build a 1D histogram of curvature information. For this, we discretized the values $c_j(p_i)$ from Eq.(3.2). Each pixel then casts a vote proportional to the gradient magnitude. Following a “soft binning” approach, it also contributes to the histograms in the four cells around it using bilinear interpolation. In practice, to calculate both the histograms of oriented gradients and histograms of curvature, the image is divided into grids of increasing resolutions for 4 levels, and histograms from each level are weighted according to $w = 2^{l-1}$, where $l = 1$ is the coarsest scale and the histograms are concatenated together to form a feature vector that encodes local and global curvature statistics of the image. The range of values from Eq.(3.2) is subdivided into 10 equally sized bins.

3.5 Model Learning

3.5.1 Support Vector Machine (SVM)

In this section we briefly review a discriminative model for classification which we use to learn our object recognition model. Support Vector Machines (SVMs) were first introduced by Vladimir N. Vapnik who together with Corina Cortes extended the idea in 1995 by using soft-margins [51]. This last approach became the standard form of this method for object recognition applications [61, 165, 80, 245].

Linear SVM

Suppose we are given a set of objects $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_m, y_m), \mathbf{x}_i \in \mathcal{H}, y_i \in \{\pm 1\}$, where $\mathcal{H}$ is any fixed dot product space and y_i are the corresponding labels (1 for positive and -1 for negative samples). Since any hyperplane in $\mathcal{H}$ is defined as

$$\{\mathbf{x} \in \mathcal{H} | < \mathbf{w} \mid \mathbf{x} >= 0\}, \ \mathbf{w} \in \mathcal{H}, b \in \mathbb{R} \tag{3.3}$$

can be transformed to its canonical form with respect to $\mathbf{x}_i, i = 1, \cdots, m$, that is

$$\min_{i=1,\cdots,m} | < \mathbf{w} \mid \mathbf{x}_i > +b| = 1, \tag{3.4}$$

the function

$$f_{\mathbf{w},b} \quad : \quad \mathcal{H} \to \{\pm 1\} \tag{3.5}$$

$$\mathbf{x} \mapsto f_{\mathbf{w},b}(\mathbf{x}) = \operatorname{sign}(< \mathbf{w} \mid \mathbf{x} > +b) \tag{3.6}$$

can be considered as a decision function[2] . The goal of a SVM consists then in finding a decision function $f_{\mathbf{w},b}$ such that

$$f_{\mathbf{w},b}(\mathbf{x}_i) = y_i, \tag{3.7}$$

(if such a function exists. The case where such a function does not exist will be treated in a later section). Since we assumed a canonical form (3.4), it follows

$$y_i(< \mathbf{w} \mid \mathbf{x}_i > +b) \geq 1. \tag{3.8}$$

Therefore, a SVM solves the following primal optimization problem

$$\min_{\mathbf{w} \in \mathcal{H}, b \in \mathbb{R}} \quad \frac{1}{2} \|\mathbf{w}\|^2, \tag{3.9}$$

$$s.t \quad y_i(< \mathbf{w} \mid \mathbf{x}_i > +b) \geq 1, \ (\forall i = 1, \cdots, m). \tag{3.10}$$

The Lagrangian formulation of the problem transforms then into

$$\max_{\alpha_i} \min_{\mathbf{w},b} L(\mathbf{w}, b, \alpha) \quad = \quad \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^{m} \alpha_i (y_i(< \mathbf{w} \mid \mathbf{x}_i > +b) - 1) \tag{3.11}$$

$$\alpha_i \geq 0. \tag{3.12}$$

Since in the saddle point, the derivatives must vanish, we obtain

$$\frac{\partial}{\partial b} L(\mathbf{w}, b, \alpha) = 0 \quad \Rightarrow \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \tag{3.13}$$

$$\frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \alpha) = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \mathbf{x}_i. \tag{3.14}$$

[2] In a 2D space this function can be understood as an indicator of the side where certain samples lie

Thus, if we substitute the solutions $\mathbf{w}, b$ into the decision function (3.5), we are able to evaluate the function only in terms of dot products taken between the input samples

$$f(\mathbf{x}) = \mathrm{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i < \mathbf{x} \mid \mathbf{x}_i > +b\right) \tag{3.15}$$

The samples $\mathbf{x}_i$, for which $\alpha_i > 0$, are called *Support Vectors* in the literature. Finally, it can be noticed that if we substitute the solutions $\mathbf{w}, b$ into the Lagrangian problem (3.11), we obtain the following formulation

$$\max_{\alpha \in \mathbb{R}^m} \quad W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j < \mathbf{x}_i \mid \mathbf{x}_j >, \tag{3.16}$$

$$s.t. \quad \alpha_i \geq 0 \; (\forall i = 1, \cdots, m), \tag{3.17}$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \tag{3.18}$$

Soft Margin SVM

In the last section, we assumed that there exists a decision function which fulfills Eq. (3.7), however, in praxis the existence of such a separating hyperplane is not guaranteed. To alleviate this problem, Cortes and Vapnik [51] proposed a different approach for the SVM. The idea behind it was to ask for an algorithm which would return a hyperplane leading to the minimal number of margin violations, that is, violations of the constraints (3.7). This was modeled by introducing the so-called slack variables

$$\xi_i \geq 0, \text{ where }, \; i = 1, \cdots, m \tag{3.19}$$

relax the constraints of (3.9) to

$$y_i(< \mathbf{w} \mid \mathbf{x}_i > +b) \geq 1 - \xi_i, \; (\forall i = 1, \cdots, m). \tag{3.20}$$

and transform the objective function in (3.9) to

$$\min_{\mathbf{w} \in \mathcal{H}, \xi \in \mathbb{R}^m} \frac{1}{2} \|\mathbf{w}\|^2 + \frac{C}{m} \sum_{i=1}^{m} \xi_i, \tag{3.21}$$

for a given constant C. It can be shown [222], that this problem results in a similar formulation as 3.16, however adding the box constraints

$$0 \leq \alpha_i \leq \frac{C}{m}. \tag{3.22}$$

Nonlinear SVM and the Histogram Intersection Kernel

Until now we only considered linear decision functions of the form sign($< \mathbf{w} \mid \mathbf{x} > +b$), however, it is possible to allow more general decision functions. To introduce this new class of SVM, the concept of *kernel* is required, which can be regarded as a generalized dot product without linearity in the arguments [222]. A kernel is a function

$$k : \mathcal{X}^2 \rightarrow \mathbb{K}, \; (\mathbb{K} = \mathbb{R} \text{ or } \mathbb{K} = \mathbb{C}) \tag{3.23}$$

from a non-empty set $\mathcal{X}$ to the real or complex numbers, which for all $m \in \mathbb{N}$ and all x_i, $i = 1, \cdots, m \in \mathcal{X}$ gives rise to a positive definite matrix $K_{ij} := k(x_i, x_j)$. The concept of a kernel is relevant for our purposes, since it can be shown ([222], sec. 2.2.2) that for any given kernel, a pre-Hilbert space (a vector space with an endowed dot product $< \cdot \mid \cdot >_B$) can be constructed such that $k(x, \hat{x}) =< \phi(x) \mid \phi(\hat{x}) >_B$, where ϕ is the projection of the data into the pre-Hilbert space. Therefore, on the practical level, due to the kernel trick ([222], s. 34), the above argumentation, and equation (3.16), the support vector Machine can be calculated directly using a kernel instead of the standard dot product of the last section. In other words, if we are interested in using a kernel in order to calculate the distances between the different objects' representations, then we can substitute the dot product used in the last section with our kernel and calculate the SVM using the kernel matrix K estimated from the training samples. Thus, the decision function in Eq. 3.15 when the dot product is substituted by the kernel yields

$$f(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + b\right) \tag{3.24}$$

For histogram-based object representations, the usage of the *histogram intersection kernel* has shown to be a useful kernel for measuring the dissimilarity between object representations ([103]). This kernel was first introduced in [229] for color indexing with an application for object recognition. If we denote with x and z two histograms, both consisting of n bins and the i-th bin is denoted with $x(i)$ and $z(i)$ respectively, then the kernel is defined as follows

$$K_{int}(x, z) = \sum_{i=1}^{n} \min\{x(i), z(i)\} \tag{3.25}$$

Furthermore, in [190] its positive definiteness was proved thus making it suitable to be used within the SVM framework.

3.5.2 Intersection Kernel SVM to Learn Curvature Representation

Because of the histogram-nature of our curvature representation (sec. 3.4), we use a histogram intersection kernel (as described in sec. 3.5.1) together with a SVM as a classifier. Specifically, in order to further accelerate our method, we utilize the SVM variant of [164], which proposed an approximation method for the Intersection Kernel SVM, which essentially reduces the runtime of the classifier to that of a linear SVM. This is done by realizing that the complexity of evaluating a SVM with histogram intersection kernel is $\mathcal{O}(mn)$ since substituting Eq. (3.25) into (3.24) yields

$$\begin{aligned} f(\mathbf{x}) &= \sum_{l=1}^{m} \alpha_l y_l \left(\sum_{i=1}^{n} \min\{x(i), x_l(i)\}\right) + b && (3.26)\\ &= \sum_{i=1}^{n} \left(\sum_{l=1}^{m} \alpha_l y_l \min\{x(i), x_l(i)\}\right) + b && (3.27)\\ &:= \sum_{i=1}^{n} \underbrace{h_i(x(i))}_{(*)} + b, && (3.28) \end{aligned}$$

where the complexity for computing each $(*)$ is $\mathcal{O}(m)$. This last computation can be reduced to $\mathcal{O}(\log m)$ by showing that it can be written as a piecewise continuous linear function

$$h_i(s) = \sum_{l=1}^{m} \bar{\alpha}_l \bar{y}_l \min\{s, \bar{x}_l(i)\} \tag{3.29}$$

$$= \sum_{1 \leq l \leq r} \bar{\alpha}_l \bar{y}_l \bar{x}_l(i) + s \sum_{r \leq l \leq m} \bar{\alpha}_l \bar{y}_l \tag{3.30}$$

$$:= A_i(r) + sB_(r), \tag{3.31}$$

where $\bar{x}_l(i)$ denotes the sorted values of $x_l(i)$ in increasing order with corresponding α's and labels $\bar{\alpha}_l$ and $\bar{y}_l$. In this case, if $s < \bar{x}_l(i)$ then $h_i(s) = 0$, otherwise r is the largest integer such that $\bar{x}_r(i) \leq s$. Both terms in function (3.31) are independent of the input data and only depend on the support vectors and α. Therefore, $h_i(\bar{x}_r)$ can be precomputed as well as $h_i(s)$ by first finding r, the position $s = x(i)$ in the sorted list $\bar{x}_i$ using binary search and interpolating between $h_i(\bar{x}_r)$ and $h_i(\bar{x}_{r+1})$. Thus, the complexity for computing $f(\mathbf{x})$ reduces to $\mathcal{O}(\log mn)$.

We train our model with an initial randomly picked subset of negative examples and then collect negative examples that are incorrectly classified by the initial model. A new model is trained using the new negative examples and the support vectors from the old model. We repeat this procedure three times. To detect an object instance the classifier is run in sliding window mode over different locations and scales. Note that using this setting, curvature does not have to be scale invariant to be used as a descriptor since the curvature computation is performed for different sizes of the sliding window, i.e. curvature is computed on different scales during detection.

3.6 Experimental Results

The objective of our experiments is to show that the direct use of curvature as a feature yields orthogonal shape information that helps to improve object detection results. Quantitatively this means that the use of our combined object descriptor should yield a higher average precision and a lower false positive rate for the same recall over the HoG descriptor using the same implementation.

We report our results on two challenging datasets: the ETHZ Shape Dataset and the INRIA horses. The ETHZ Shape Dataset contains 255 images belonging to five different classes. We follow the standard experimental protocol for creating training and test sets. The the INRIA horses dataset consists of 170 images containing one or more side-viewed horses and 170 images without horses. 50 horse images and 50 negative images are used for training and the remaining 120 horse images plus 120 negative images are used for testing. In our experiments we are following the standard PASCAL setting for counting true positives and false positives among the predicted bounding boxes. In table 3.1 we compare the performance of our approach with several state-of-the-art detector systems [165, 107, 228] at 0.3, 0.4 and (for the INRIA horses) 1 FPPI. Our HoG baseline implementation uses HoG and IKSVM, like the currently best reported results of a HoG based detection system on ETHZ [165]. Note that [165] searched over different aspect ratios for some categories in the ETHZ Shape Dataset (e.g. Giraffes and Mugs). This explains the differences in the baseline results (HoG vs. IKSVM). Our final detector HoGC clearly improves performance over the baseline HoG detection system on both datasets. Furthermore, our approach outperforms the voting approach suggested in [107]. In addition, we compared our detection system with the descriptive shape model (DSM) suggested in [228]. This approach performs slightly better than our HoGC descriptor on the ETHZ Shape dataset since it also adds a deformable part model to the holistic approach. As reported in [1] the average performance improves about 8% on PASCAL VOC 2007 when adding part-based HoG descriptors. However, we decided for a fair comparison with HoG implementations to use the standard setting without parts. Furthermore, detection takes several minutes per image using the descriptive shape model, whereas using HoGC is one order of magnitude faster.

Figures 3.3 and 3.4 compare our approach with the state-of-the-art HoG detector. We remark that the authors in [165] did not include FPPI or precision-recall curves for their IKSVM + HoG detector for the ETHZ Shape Dataset. By incorporating curvature information, our combined HoGC representation outperforms HoG results in all categories of the ETHZ Shape Dataset and on the INRIA horses. We achieve an average gain of 7.6% in AP on the ETHZ Shape Dataset and of 12.3% on the INRIA horses. For the ETHZ Shape Dataset we get on average a 5.4% higher detection rate at 0.3 FPPI and at 0.4 FPPI; an improvement of 7%. On the INRIA horses we improved the recall by 8.7% at 0.3 FPPI, 7.6% at 0.4 FPPI and 3.2% at 1 FPPI. For the sake of completeness, we also included detection results of our system solely using curvature information. However, the suggested curvature feature was never intended to be used in solitude and for that reason does not contain redundant information of the HoG descriptor, like the orientation of curvature. That explains the drop in performance when using curvature without HoG while the combination of both significantly improves state-of-the-art HoG object detection methods. These results approve our initial hypothesis that curvature is a complimentary feature to HoG.

3.7 Discussion

The main contribution of this chapter is to provide quantitative evidence that curvature information of objects can be discriminatively used in a robust and reliable manner for object recognition. Our results show that the use of curvature information yields orthogonal information to the state-of-the-art theme of histograms of oriented gradients for visual search tasks. Combining both leads to improved accuracy and performance on standard datasets and significantly improves the state-of-the-art detection system solely based on HoG. The proposed curvature-based object representation is generic, efficient to compute, and it can be effortlessly integrated into all current object models that utilize histograms of gradients. Thus a wide applicability is automatically granted.

Table 3.1: We compare the performance of the HoGC against the state-of-the-art detector IKSVM [165] for the **ETHZ Shape Dataset**. We follow the standard setup of HoG and search over location and scale, but not over aspect ratios. This explains the performance gap between our HoG and IKSVM [165] on ETHZ. [228] deviate from HoG by adding a computationally costly part-based model .

	ETHZ Shape: Average Precision		
	Curv.	**HoG**	**HoGC**
Applelogos	72.3	86.7	92.5
Bottles	72.0	79.0	88.4
Giraffes	31.0	56.0	60.1
Mugs	34.1	71.2	82.2
Swans	50.2	59.4	66.9
Average	52.1	**70.4**	**78.0**

Table 3.2: We compare the performance of the HoGC against the state-of-the-art detector IKSVM [165] for the **the INRIA horses dataset**.

INRIA Horses: Average Precision		
Curv.	**HoG**	**HoGC**
52.2	**71.3**	**83.6**

Table 3.3: ETHZ Shape Dataset: False positives per image at 0.3, 0.4 and 1 recall. We follow the standard setup of HoG and search over location and scale

	ETHZ Shape: Recall @ 0.3/0.4/(1) FPPI					
	Curvature	**HoG**	IKSVM [165]	Voting [107]	DSM [228]	**HoGC**
Applelogos	86.3/91.2	90.0/90.0	90.0/90.0	90.6±6.2/-	95.0/95.0	100/100
Bottles	92.8/96.4	96.3/96.3	96.4/96.4	94.8±3.6/-	100/100	96.4/96.4
Giraffes	43.0/43.0	72.3/78.7	79.1/83.3	79.8±1.8/-	87.2/89.6	74.4/85.1
Mugs	54.8/54.8	87.1/87.1	83.9/83.9	83.2±5.5/-	93.6/93.6	90.3/93.5
Swans	76.4/76.4	82.3/82.3	88.2/88.2	86.8±8.9/-	100/100	94.1/94.1
Average	70.6/72.3	**85.6/86.8**	87.5/88.4	87.1±2.8/-	95.2/95.6	**91.0/93.8**

Table 3.4: INRIA Horses: False positives per image at 0.3, 0.4 and 1 recall.

INRIA Horses: Recall @ 0.3/0.4/(1) FPPI					
Curvature	**HoG**	IKSVM [165]	Voting [107]	DSM [228]	**HoGC**
53.2/56.5/72.8	**81.5/82.6/91.3**	-/-/86.0	-/-/-	-/-/-	**90.2/90.2/94.5**

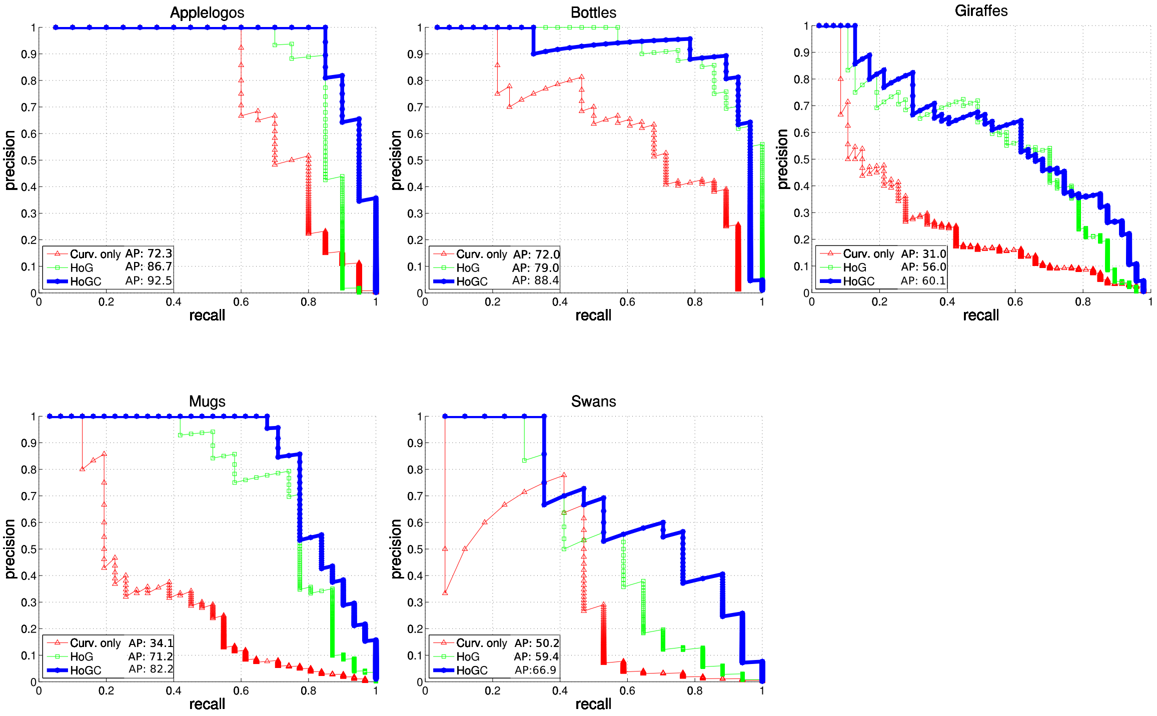


Figure 3.3: Precision Recall Curves for ETHZ Shape dataset comparing curvature only (red), HoG (green) and HoGC (blue).

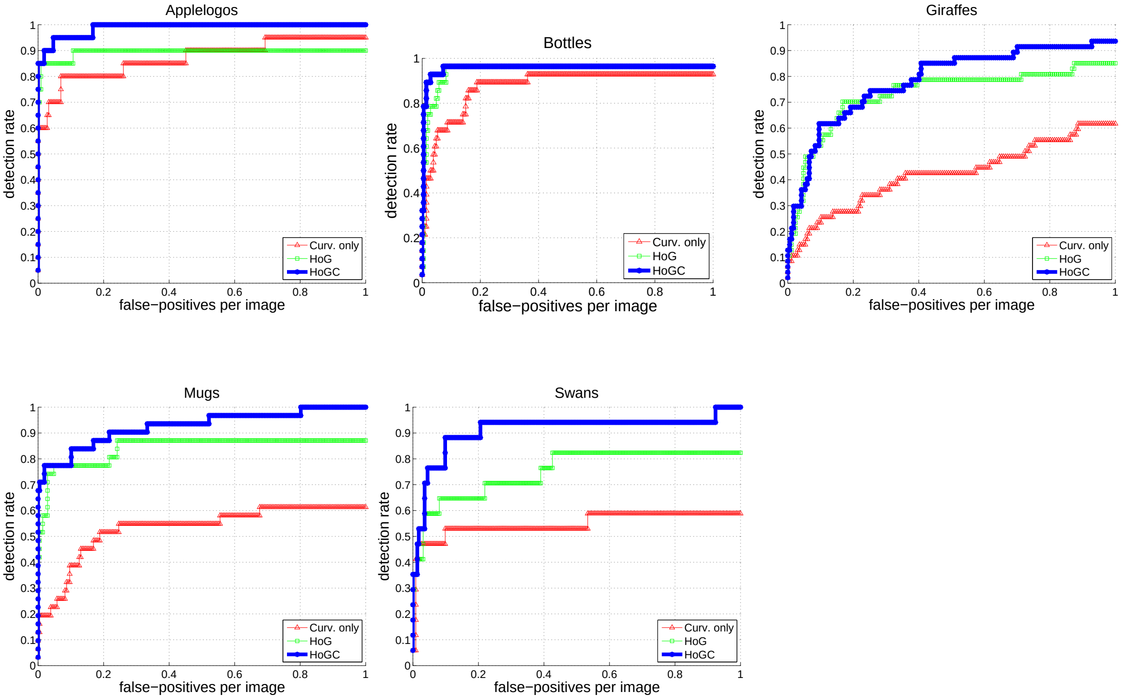


Figure 3.4: Detection performance against FPPI for the ETHZ Shape dataset comparing curvature only (red), HoG (green) and HoGC (blue).

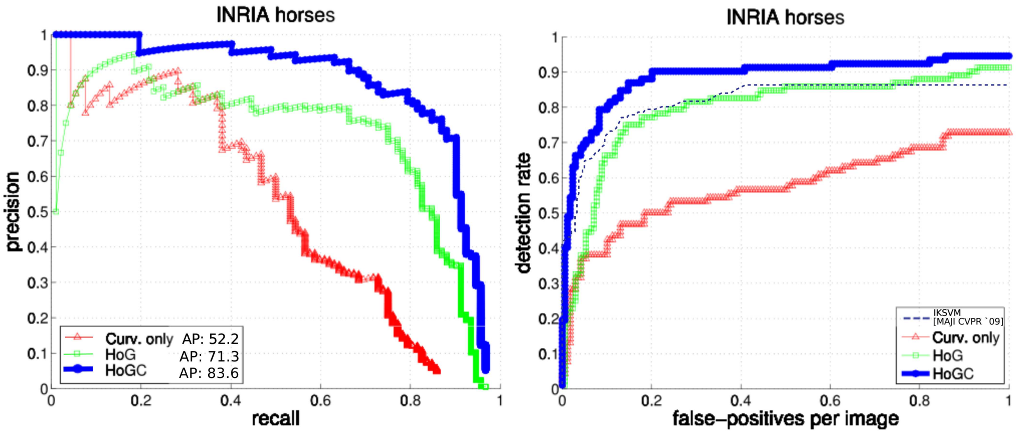
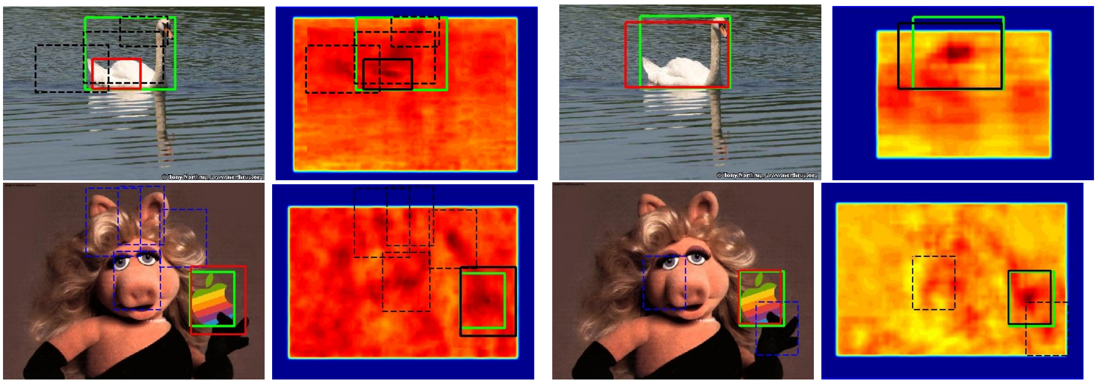


Figure 3.5: Precision Recall Curve and detection performance against FPPI for the the INRIA horses dataset; comparing curvature only (red), HoG (green) and HoGC (blue).



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

Figure 3.6: Detection results using standard HoG (implementation of [165]) (first two columns) and results using HoGC (last two columns). First detection is outlined in red and false positives in dashed black. These examples illustrate a general finding in this database that compared to the widely used HoG, our proposed representation yields a better localization of the maxima compared to ground-truth and generation of less false-positives.

CHAPTER 4

BEYOND BOUNDING-BOXES: LEARNING OBJECT SHAPE BY MODEL-DRIVEN GROUPING

4.1 Going Beyond Bounding-Boxes

As seen in chapter 2, object detection typically proceeds by localizing object bounding-boxes (e.g. [80]), which are parameterized by their location, scale, and aspect ratio. A classifier is then evaluated for each detection window, thereby providing hypotheses that are ranked by their score. Such approaches have proven to be very successful for benchmarks, but there are two issues that remain unresolved. First, objects are not box-shaped and so the detection window contains a significant amount of background clutter that tends to deteriorate the whole window's classification result. And indeed, even complex models like [80] are eventually based on a holistic representation of the whole bounding-box, including the clutter. The second problem is that the object shape becomes only available for detection once the object has been segregated from the background. To overcome both problems, not only background suppression is required, but also reasoning about the object shape is essential. Recent work (e.g. [151]) in the field of segmentation has shown that relying only on low-level cues is not enough. Furthermore, it appears reasonable to combine class-specific top-down information to achieve better results. The purpose of this chapter is to learn object models for detection by explicitly representing object shape and segregating it from the background, without, however, requiring manual segmentation of the training samples. Therefore, we propose a model-based approach that does not require supervision, but automatically learns object shape and appearance while segregating objects from the background. Since we use more than a mere bottom-up segmentation, we are able to capture the overall object shape in a model-driven manner by grouping the corresponding foreground regions.

Finally, in section 1.5.1 we observed how an object recognition system can be understood as an abstraction process or mapping between the image level and the space of object representations. From this perspective, the method introduced in this section corroborates the observation that using a richer description of the object's shape in the underlying class-model of the system helps to improve the overall detection performance of the system. The presented results were published in [184]

4.2 Novelty of the Approach

The present approach constitutes an advance with respect to the state-of-the-art. This can be seen by first considering supervised methods like ([99, 172, 247]). The disadvantage of these methods is that they require ground-truth pixel-wise segmentation masks during training. Such information is usually not available for large-scale detection tasks (s. section 2.3.3) or is tedious and expensive to obtain, so we are proposing an automatic MIL learning-based (s. section 4.5.2) approach to circumvent these shortcomings. On the other hand, we have methods which only require bounding-box information during training [166, 232, 254, 41, 37, 106, 243]. These methods differ in the way shape information is integrated into the detection task. Pure bottom-up methods [166, 232] are susceptible to segmentation artifacts. While [166] directly classifies bottom-up generated segments using a k-nearest neighbor classifier, [232] computes hierarchical segmentations to find object subtrees similar to those learned during training. [254, 41] can be viewed as top-down approaches. [254] divides the bounding-box into cells and infers an occlusion map by clustering the response scores of a linear SVM on each cell, where occluded regions are defined as the groups with a negative overall response. This approach does not use any shape information to train the linear SVM. Furthermore, negative response scores can also be caused by occlusion or by other factors, such as background or an uncommon shape. Based on the model of [273], [41] attempts to capture the object's shape by means of a fixed number of coarse box-shaped patterns. Finally, methods like [106, 37, 243] attempt to combine bottom-up and top-down cues. For instance, Gu et al. [106] proposed a method for detection using regions. Starting with regions as the basic elements, a generalized Hough-like voting strategy for generating hypotheses is used (see [194] for improvements to the idea of voting). The method's drawbacks are twofold. First, it needs a general sliding window classifier for verification, which does not take shape into account. Second, ground-truth pixel-wise segmentation masks for the training data are required. Recently, [37] proposed a method for object detection based on the category-independent figure-ground segmentation masks of [38]. To train with only bounding-box information, the authors assume that the best ranked segment within the bounding-box covers the entire object. This segment is thus used to learn a regression function that predicts the quality of query segments. Consequently, the performance of their detection system is highly susceptible to the fact that the first bottom-up generated segment actually covers the entire object. In datasets like PASCAL VOC we observe that in many images this assumption is too strong. Finally, [243] utilizes multiple over-segmentations to propose class-independent bounding-boxes for classification. However, the authors discard the shape information contained in the super-pixels. They sample at each pixel 5 different color features and utilize them within a standard bag-of-words model to classify the object. Thus, the question of using the foreground object shape for object detection remains open.

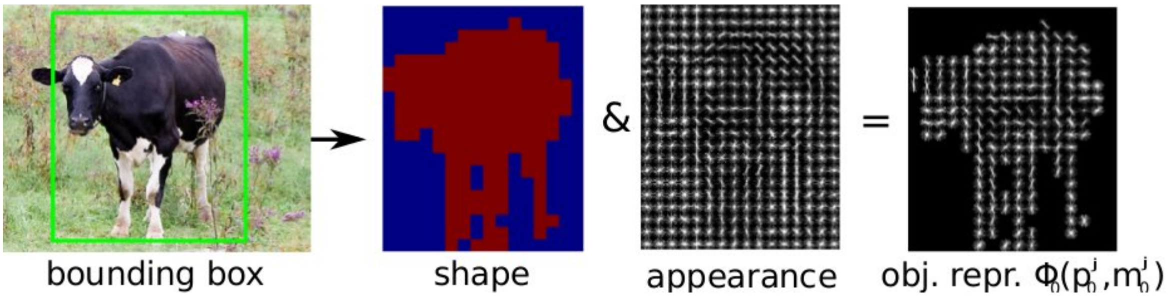


Figure 4.1: Object representation (best viewed in color). We divide the bounding-box into cells and calculate features on each cell. Inferring a foreground segmentation cell-mask from unsegmented training data, we suppress the background features by setting the corresponding cells to zero.

4.3 Suppressing the Background

Detection window approaches like [60, 80] have demonstrated a good performance in difficult benchmarks. Consequently, such a framework offers us a good basis to implement our idea. The detection window is commonly divided into a grid of cells and we learn object shape in order to suppress cells in the clutter and concentrate on the actual object. In this section we describe how to model a foreground/background segregation.

Suppose an object $\mathcal{O}^j$ within image I is given and we assume for a moment a pixel-wise foreground object's segmentation is also given. In the next section we will describe how to automatically learn a cell-accurate shape estimation for the object's foreground.

First, we divide the bounding-box j into an array of size $l_0 \times h_0$. For each cell we calculate a $d-$dimensional feature. This $l_0 \times h_0 \times d$ matrix is called $\hat{\phi}_0(p_0^j)$, where $p_0^j = (x, y)$ is the top-left position of the bounding-box in image I. Specifically, in this chapter we use histograms of oriented gradients (HoG) as features. These widely used and fast to calculate descriptors capture the edge or gradient structure that is very characteristic of local shape. Additionally, they exhibit invariance to local geometric and photometric transformations ([80, 60]). However, our framework is independent of this specific choice of features. A combination of different descriptors (e.g. like in [243]) can be integrated into our model and should enable further performance improvements.

The foreground of an object is modeled by defining a binary vector $m_0^j \in \mathbb{B}^{1\times l_0 h_0}$. This vector contains ones if the corresponding cell is covered by the object, otherwise it is zero. We call this vector m_0^j the *root-cell mask* for object $\mathcal{O}^j$ (part-cell masks are introduced in Sec. 4.8). Using m_0^j we set to zero the cells of $\hat{\phi}_0(p_0^j)$ corresponding to the zero entries in m_0^j. Formally, the foreground representation of object $\mathcal{O}^j$ is defined as

$$\phi_0(p_0^j, m_0^j) := (m_0^j \otimes \mathbf{1}_d) \odot \hat{\phi}_0(p_0^j), \tag{4.1}$$

where $\otimes$ defines the Hadamard-Product and $\odot$ the element-wise multiplication. Fig. 4.1 shows how to suppress the background of a bounding-box if a root-cell mask for the object's foreground is given.

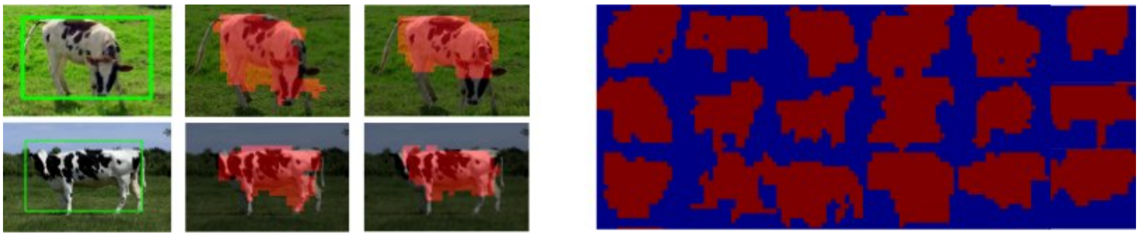


Figure 4.2: Left: the first column shows a detection and the last two columns the two most similar prototypical segments. Right: Subset of prototypical segments for the category cow.

4.4 Matching Objects

Due to different pose variations, occlusion and clutter, the foreground root-cell masks m_0^j and m_0^u of two objects may differ substantially. Therefore, building an Euclidean dot product between the feature representations $\phi_0(p_0^j, m_0^j)$ and $\phi_0(p_0^u, m_0^u)$ as [99] or [80] do, will lead to unstable matching scores. Rather than using a simple dot product, we represent each object with a prototypical set of shape segments $C_0 = \{\bar{m}_0^\iota\}_{\iota=1}^{\nu}$. This set of segments is automatically learned from unsegmented training data (see section 4.7 for more details). The idea is to reduce the high intra-class shape variability by using a reduced number of typical class-specific views of its shape. We then use a weighted sum to match both representations. Precisely, the matching score is given by

$$
\begin{aligned}
d_0(\phi_0(p_0^j, m_0^j), \quad & \phi_0(p_0^u, m_0^u)) := \\
& \frac{1}{\nu}\sum_{\iota=1}^{\nu} < a(m_0^j, \bar{m}_0^\iota)\phi_0(p_0^j, \bar{m}_0^\iota), a(m_0^u, \bar{m}_0^\iota)\phi_0(p_0^u, \bar{m}_0^\iota) >, \qquad (4.2)
\end{aligned}
$$

where

$$
a(m_0^j, \bar{m}_0^\iota) := exp\left(-\beta * \frac{\|m_0^j - \bar{m}_0^\iota\|_2}{|\bar{m}_0^\iota|}\right) \tag{4.3}
$$

represents the dissimilarity score between the root-cell mask m_0^j and the prototypical root-cell mask $\bar{m}_0^\iota$. The parameter β is obtained by cross-validation. In our experiments, we obtained an optimal value in the range of 1.1 ± 0.1 for the different object classes. Here $|\bar{m}_0^\iota|$ represents the total number of active cells in the prototypical root-cell mask $\bar{m}_0^\iota$. Equation (4.2) induces a Mercer kernel, since the sum of Mercer kernels is a Mercer kernel again. By the "Kernel Trick" we know, that there exists a (possibly unknown) transformation Φ into a space in which the kernel (4.2) is a scalar product. To keep the notation simple, we identify $\mathcal{O}^j := \Phi(\phi_0(p_0^j, m_0^j))$ and refer to this scalar product as

$$
< \mathcal{O}^j, \mathcal{O}^u >_{CB} := d_0(\phi_0(p_0^j, m_0^j), \phi_0(p_0^u, m_0^u)). \tag{4.4}
$$

In praxis we do not need to evaluate the function Φ to learn our model, but use the kernel values instead. By defining the kernel (4.2) we have integrated both of our goals into the detection window approach: We suppress the features corresponding to the background and robustly represent the shape of an object through a prototypical set of shapes.

Let us assume for the moment that for all objects $\mathcal{O}^j$ in the training data with their root-cell masks m_0^j containing the whole object foreground are given. The training set is denoted by $\{(\mathcal{O}^j, y^j)\}$. Here $y^j \in \{1, -1\}$ denotes the label of object $\mathcal{O}^j = \Phi(\phi_0(p_0^j, m_0^j))$. In this special case, we could easily learn a discriminative function

$$f(\phi_0(p_0^q, m_0^q)) = \sum_{i \in SV} -y_i \alpha_i d_0(\phi_0(p_0^q, m_0^q), \phi_0(p_0^i, m_0^i)) + b \tag{4.5}$$

to classify the query object $\phi_0(p_0^q, m_0^q)$ (SV is the set of support vectors). However, in contrast to [99], we are not provided with the foreground root-cell masks m_0^j during training, but rather we automatically learn them from unsegmented training data. This is described in the next section. Similar to [99], [37] assumes that the best-ranked foreground segmentation mask of [38] covers the whole object. In practice this assumption is, however, not valid: The second row of figure 4.3 shows the best ranked CMPC segments that lie within the object bounding-box. None of them covers the whole object exactly.

4.5 Learning from Unsegmented Training Data

Before we introduce our approach it is necessary to first introduce the underlying general learning paradigm which is called Multiple Instance Learning or MIL. This will be done in the next subsection and thereafter we will describe how to use this paradigm in our context.

4.5.1 Multiple Instance Learning (MIL) paradigm

The underlying idea behind MIL consists of a variation of the classical supervised learning task (s. SVM in section 3.5.1). Supervised discriminative learning algorithms infer a decision function (classifier) from labeled training data pairs $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i$ is the input pattern and y_i its corresponding label. Instead of considering pairs $(\mathbf{x}_i, y_i)$, MIL algorithms receive as input sets or *bags* containing several instances and labels for each of the bags. A bag is commonly considered positive if all instances within it are positive, and as negative if at least one instance within the bag is negative ([67, 2]). The task of the learned classifier is either (a) to infer the label of all instances within a test bag or (b) infer the label of the bag without inferring the label of the contained instances. The general idea of MIL was first proposed by Dietrich et al. [67], where axis-parallel rectangles bounding positive examples were learned to classify the bags (i.e. this approaches solves task a). Since the introduction of SVMs for solving the MIL task by Andrews et al. [2], several other algorithms have been proposed in literature [66, 100, 32, 272, 252]. However, the main idea can be understood considering two variants of SVM-based MIL learning methods introduced by Andrews et al. in [2]. Both methods are called MI-SVM and mi-SVM which we briefly review in the following. For simplicity we will formulate both approaches using *linear* SVMs as described in section 3.5.1. However also nonlinear kernels can be used for solving the task.
Given a set of input patterns $\mathbf{x}_1, \cdots, \mathbf{x}_m$ grouped into bags $\mathbf{B}_1, \cdots, \mathbf{B}_n$, with

$$\mathbf{B}_I := \{\mathbf{x}_i \, : \, i \in I\} \tag{4.6}$$

for given index sets $I \subseteq \{1, \cdots, m\}$. A bag-label Y_I is associated with each bag $\mathbf{B}_I$ and the relation between instance label y_i and bag-labels can be expressed as $Y_I = \max_{i \in I} y_i$

or alternatively as a set of linear constraints

$$\begin{aligned}\sum_{i \in I} \frac{y_i + 1}{2} &\geq 1, \; \forall I \, s.t. \, Y_I = 1 \\ y_i &= -1, \; \forall I \, s.t. \, Y_I = -1\end{aligned}$$

mi-SVM

The mi-SVM formulation

$$\min_{y_i} \min_{\mathbf{w},b,\xi} \frac{1}{2}\|w\|^2 + C \sum_i \xi_i \tag{4.7}$$

$$s.t. \quad \forall i : y_i(< \mathbf{w} \mid \mathbf{x}_i > +b) \geq 1 - \xi_i, \; \xi_i \geq 0, \; y_i \in \{-1, 1\} \tag{4.8}$$

$$\sum_{i \in I} \frac{y_i + 1}{2} \geq 1, \; \forall I, \; s.t. \, Y_I = 1 \tag{4.9}$$

$$y_i = -1, \; \forall I \, s.t. \, Y_I = -1 \tag{4.10}$$

treats the labels y_i of $\mathbf{x}_i$ belonging to a positive bag as unknown integer variables. Therefore, the mi-SVM formulation maximizes a soft-margin together with label assignments as well as hyperplanes, leading to a mixed-integer program. This differs from the standard SVM formulation (section 3.5.1), where labels y_i for all instances are known and the problem reduces to the hyperplane estimation.

MI-SVM

This formulation is an alternative to mi-SVM and extends the notion of margin from individual instances to bags. Here, the prediction for a bag takes the form

$$\hat{Y}_I = \operatorname{sgn} \max_{i \in I} \left(< \mathbf{w} \mid \mathbf{x}_i > +b\right), \tag{4.11}$$

where $\max_{i \in I} (< \mathbf{w} \mid \mathbf{x}_i > +b)$ can be seen as a generalization of the margin for bags. In this formulation, only one pattern per positive bag matters, since it will define the bag-margin. Once this instance is identified, the position of all other instances in the bag become irrelevant. Using this notation, the MI-SVM version of MIL is defined as

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_I \xi_I \tag{4.12}$$

$$s.t. \quad \forall I : \, Y_I \max_{i \in I}(< \mathbf{w} \mid \mathbf{x}_i > +b) \geq 1 - \xi_I, \, \xi_I \geq 0. \tag{4.13}$$

It is worth noting that in this case only slack variables for bags are defined. Furthermore, the MI-SVM formulation can also be cast as a mixed-integer program. For positive bags an integer variable $1 \leq z(I) \leq |\mathbf{B}_I|$ is used to indicate the ”most positive“ member

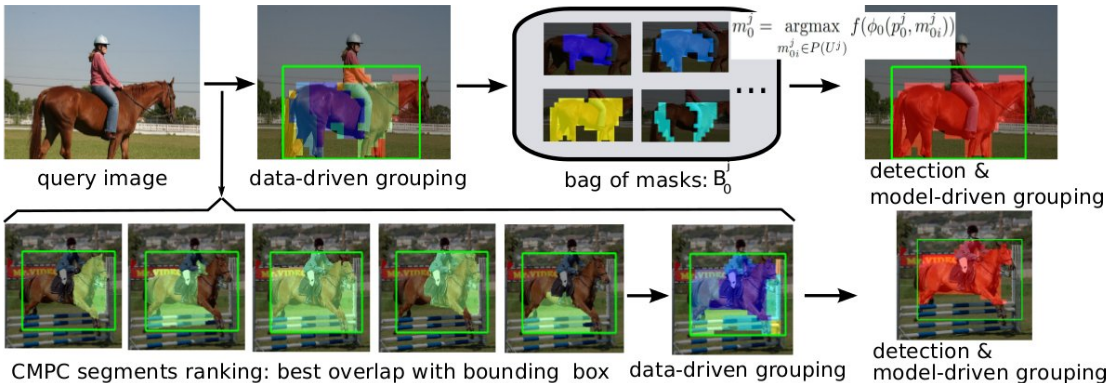


Figure 4.3: First row: We simultaneously detect and infer the object foreground. Second row: We show our data-driven grouping from which we infer the foreground of our object. For complex categories we cannot assume that the first CMPC segment covers the whole object.

$\mathbf{x}_{z(I)} \in \mathbf{B}_I$ and thus the constraint (4.13) reduces to $< \mathbf{w} \mid \mathbf{x}_{z(I)} > +b \geq 1 - \xi_I$ resulting in the formulation

$$\min_{z} \min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{I=1} \xi_I \tag{4.14}$$

$$s.t. \quad Y_I(< \mathbf{w} \mid \mathbf{x}_{z(I)} > +b) \geq 1 - \xi_I, \ (\forall I) \tag{4.15}$$

$$(< \mathbf{w} \mid \mathbf{x}_{z(I)} > +b) \geq (< \mathbf{w} \mid \mathbf{x} > +b), \ \forall x \in \mathbf{B}_I, I \tag{4.16}$$

$$\xi_I \geq, \forall I \tag{4.17}$$

where constraint (4.16) is added to enforce that $\mathbf{x}_{z(I)} \in \mathbf{B}_I$ is the "most" positive member of the bag. Whereas in this formulation instances belonging to negative bags do not have any impact on the objective function, positive bags $\mathbf{B}_I$ are represented by a single instance $\mathbf{x}_{z(I)} \in \mathbf{B}_I$.
Both formulations MI-SVM and mi-SVM also hold for the general case, where a kernel function is used (s. section 3.5.1).
In order to solve the program 4.14, the authors in [2] proposed a two stage greedy algorithm, which alternates the following steps:

- for given $z(I)$ $(\forall I : Y_I = 1)$, solve the resulting SVM to estimate the optimal hyperplane
- for a given hyperplane, update the integer variables $z(I)$ in a way that the objective function is locally minimized.

The last step is run for each bag independently choosing the instance which reduces the objective function the most.

4.5.2 Our Model: Using MIL to Find Foreground Masks

The question now is how to learn the classification function f if the foreground root-cell masks m_0^j are not given during training?

Given a discriminatively trained function f, the problem of inferring the foreground root-cell mask m_0^j for an object $\mathcal{O}^j$ can be formulated as

$$m_0^j = \underset{m_0}{\operatorname{argmax}} \, f(\phi_0(p_0^j, m_0)) \tag{4.18}$$

i.e. the inference (4.18) is tackled by grouping cells in a model-driven, top-down manner so as to maximize the classification score.

We simultaneously learn the function f and solve the grouping problem by formulating our problem in the Multiple Instance Learning (MIL) framework. Here, a bag contains features corresponding to different root-cell masks. For positive instances, at least one of these features corresponds to the foreground of an object. In the ideal case, a bag B_0^j would contain all possible combinations of cells within the bounding-box. Since this is not tractable, in the next section we describe how to create a shortlist of meaningful groups in a bottom-up manner. Suppose we obtain l different groups for a bounding-box j. The i-th group is represented by a root cell mask m_{0i}^j and build the set $U^j := \{m_{0i}^j\}_{i=1}^l$. A bag is then defined as

$$B_0^j := \left\{\phi_0(p_0^j, m_{0i}^j) | m_{0i}^j \in P(U^j)\right\}_{i=1}^{|P(U^j)|}, \tag{4.19}$$

where $P(U^j)$ is the power set of U^j. If the bounding-box contains an object, the label Y_j of the bag B_0^j is set to 1, otherwise it is -1. Using our kernel (4.2) the problem of learning the function f transforms into:

$$\min_{w_0, b, \xi} \frac{1}{2} \|w_0\| + C \sum_I \xi_I \tag{4.20}$$

$$s.t. \qquad \forall I : Y_I \max_{i \in I} (< w_0, \mathcal{O}_i^I >_{CB} + b) \geq 1 - \xi_I, \xi_I \geq 0, \tag{4.21}$$

here $\mathcal{O}_i^I = \Phi(\phi_0(p_0^I, m_{0i}^I))$ are object hypotheses and denote the elements within the bag I. Once the function f is learned, the inference problem (4.18) for a query image is transformed into

$$m_0^j = \underset{m_{0i}^j \in P(U^j)}{\operatorname{argmax}} \; f(\phi_0(p_0^j, m_{0i}^j)) \tag{4.22}$$

In other words, in (4.22) we look for the "most" positive instance within B_0^j and by doing this, we indirectly infer the corresponding root-cell segmentation mask m_0^j (s. first row of Fig. 4.3). In practice the optimization problem (4.20) is solved using the MI-SVM formulation of Andrews [2] described in the last section (s. section 4.5). Specifically, the calculation of the hyperplane w_0 and bias b is alternated with the calculation of the margin for the positive bags: $Y_I \max_{i \in I}(< w_0, \mathcal{O}_i^I >_{CB} + b)$. This means that for every positive bag we fix the "most" positive instance and then we use all other instances of the negative bags to learn a SVM using our Mercer kernel (4.2). In our experiments we used this MIL formulation since it is effective, fast (convergence is reached after a few iterations) and the performance was robust for varying initializations. Specifically, we randomly chose an element for every positive bag to initialize the algorithm.

In the first row of Fig. 4.3 we visualize the inference of the final foreground root-cell mask m_0^j, given a data-driven grouping of cells for the bounding-box.

4.6 Data-driven Grouping

In this section we describe how to create a shortlist of candidate groups by means of a data-driven grouping of cells for a given bounding-box. This is necessary to render the inference problem (4.18) and the creation of bags (4.19) feasible.

Recently, [38] presented the combinatorial CMPC algorithm for generating a set of binary figure-ground segmentation hypotheses $\{S_t^I\}_{t=1}^{N_s}$ for an image I. In general, we can not assume (see second row of Fig. 4.3) that the best ranked segment covers the whole object (as in [37]). However, the pool of CMPC segments yields a good basis to obtain groups of pixels, which cover only parts of the object. An example of our grouping can be seen in Fig. 4.3 (second row).

Given a bounding-box BB_j in image I, the idea is to first weight each pixel-wise segment S_t^I generated by [38] with the ratio between the number of pixels p_{kl} belonging to the segment S_t^I which lie outside the bounding-box and the total number of pixels covered by the bounding-box $|BB_j|$ itself:

$$r_t^j := \frac{1}{|BB_j|} \sum_{kl} \mathbb{1}_{[p_{kl} \in S_t^I]} \mathbb{1}_{[p_{kl} \notin BB_j]}. \tag{4.23}$$

Only segments S_t^I that fully lie within the bounding-box will get high scores, while straddling segments will be penalized. We then take the weighted sum of all segments which intersect the bounding-box and build a density map for this bounding-box

$$\mathcal{H}_{kl}^j := \frac{1}{N_s} \sum_t^{N_s} r_t^j * \mathbb{1}_{[p_{kl} \in S_t^I]} \mathbb{1}_{[p_{kl} \in BB_j]}. \tag{4.24}$$

The values in this map indicate which regions within the bounding-box were consistently covered by CMPC segments S_t^I. We then apply a mean-shift clustering algorithm on this 2D density map $\mathcal{H}^j$ and enforce the connectedness of each of the resulting groups. The cells covering each of these groups define the root-cell masks m_{0i}^j, used to construct the bags in equation (4.19).

In practice, for bounding boxes containing an object, we typically obtain between 6 and 8 groups. For boxes in the background, our grouping algorithm typically does not generate any segment, since these regions are not covered by a CMPC segment (the weights in Eq. 4.23 are zero). This situation renders an exhaustive search using inference (4.18) feasible. We favor mean-shift over other clustering methods because it allows an adaptive bandwidth for different clusters.

4.7 Learning a Prototypical Set of Segments

Our goal is to represent every object through a prototypical set of segments. In section 4.4 we used such a representation to robustly match different object instances. In this section we describe how to learn such a prototypical set.

The idea is to explain the shape complexity of a class through a reduced number of segments that are typical for a certain class. Using the bottom-up grouping described in section 4.4, we obtain a bag B_0^j for every positive training sample j. To find those specific segments that appear frequently within the class, we hierarchically cluster the elements of all positive bags (e.g. using Ward's method). Every group is then represented by its medoid, i.e. the element with the minimal average dissimilarity (using measure (4.3)) to all the objects in the cluster. The set of all medoids define the prototypical set of segments $C_0 = \{\bar{m}_0^\iota\}_{\iota=1}^{\nu}$ used to train our model. The number of clusters is chosen using cross-validation and ranges between 10 and 40 segments (s. Fig. 4.2).

4.8 Implementation Details

We use a sliding window detection model similar to [80] to implement our idea. The model in [80] describes an object $\mathcal{O}^j$ by means of a bounding-box covering the entire object (root window) as well as eight smaller windows (about half the size) that cover parts of the root window. Every part window i is divided into a grid of cells of size $l_i \times h_i,\ i = 1 \dots 8$ and a HoG feature is calculated for every cell. During training, weights (used as linear filters) are learned for the root window and additional 8 linear filters are trained for the parts. In our case, if we ignore the parts for a moment, we first would need to learn the prototypical set of segments using the positive training samples as described in Sec. 4.7 and then learn the classifier (Eq. 4.5) as described in Sec. 4.5.2. To include the concept of parts from [80], we will first introduce the notion of a bag for each of the part windows and then extend our matching kernel (4.2) for these parts also. Thereafter, the corresponding classifier can be trained analogously to [80] and thus we remit to that work for further details.
Modeling parts: Running the bottom-up grouping described in section 4.7 exclusively on the root window, results in the bags B_0^j for each training sample j (τ being the number of instances in each bag). We then define a bag B_i^j for each of the part-windows as follows:

$$B_i^j := \{\phi_i(p_i^j, m_{ik}^j)\}_{k=1}^{\tau},\ i = 1 \dots 8. \tag{4.25}$$

Here p_i^j denotes the position of the i-th part-window for sample j. The binary vector $m_{ik}^j \in \mathbb{B}^{l_i h_i}$ denotes the k-th part-cell segmentation mask of part i. It is obtained by taking the overlap of part i with the root-cell mask $m_{0k}^j \in \mathbb{B}^{l_0 h_0}$. In doing so, we obtain the feature representation $\phi_i(p_i^j, m_{ik}^j)$ for the i-th part (similar to Eq. (4.1)). Following this notation, the matching score of Eq. 4.2 between two objects $\mathcal{O}^j, \mathcal{O}^u$ can be extended to include parts,

$$d(\mathcal{O}^j, \mathcal{O}^u) := \sum_{i=0}^{8} d_i(\phi_i(p_i^j, m_i^j), \phi_i(p_i^u, m_i^u)) + < p_i^j - p_0^j, p_i^u - p_0^u > . \tag{4.26}$$

Here the last term compares the displacement of the i-th part w.r.t. the object center. $d_i(.,.)$ denotes the matching score for part i defined as in Eq. (4.2). To obtain $d_i(.,.)$ we also use a set of prototypical segments to represent each one of the parts. This set is obtained in a similar way as for the root window by hierarchically clustering the elements of all positive bags B_i^j. In practice, 7 prototypical segments are used to represent each part. Using the kernel (4.26) we are then capable of learning a discriminative function along the lines of (4.5).

4.9 Experimental results

The purpose of our experiments is to show that if only bounding-box annotated data is available during training, using a top-down generated prototypical representation of the

Table 4.1: Detection results for the ETHZ-Shape dataset. Performance is measured as *pixel-wise* AP over 5 trials, following [37, 106]. For completeness, we include the performance of [80] measured using a bounding-box parametrization. We improve the state-of-the-art by 7% AP

	Our Method	Carreira etal. [37]	Gu etal. [106]	Felz. etal.[80]
Apples	0.963 ± 0.023	0.890 ± 0.019	0.772 ± 0.112	0.934 ± 0.048
Bottles	0.877 ± 0.011	0.900 ± 0.021	0.906 ± 0.015	0.891 ± 0.028
Giraffes	0.823 ± 0.038	0.754 ± 0.019	0.742 ± 0.025	0.817 ± 0.048
Mugs	0.885 ± 0.037	0.777 ± 0.059	0.760 ± 0.044	0.856 ± 0.073
Swans	0.927 ± 0.023	0.805 ± 0.028	0.606 ± 0.013	0.813 ± 0.125
Mean	$\mathbf{0.896} \pm 0.026$	0.825 ± 0.012	0.757 ± 0.032	0.862 ± 0.051

object shape, as well as suppressing the background within a bounding-box, helps to improve pixel-wise object detection.
The methods of [37] and [106] are the most similar to ours and therefore provide us with a baseline for our results. Both methods present pixel-wise detection results exclusively on the ETHZ-Shape dataset ([90]). Specifically, [37] also presents results for the PASCAL segmentation challenge. However, this challenge assesses a simpler problem than that in our method since pixel-wise segmentation masks are used for training the model. For purposes of comparison with state-of-the-art [37, 106] we also use the ETHZ-Shape dataset to test our model's performance. Larger and more complex datasets for object detection (e.g. INRIA Horses or PASCAL VOC) are suboptimal to demonstrate the ability of our method, since there are no pixel-wise masks for the whole test-set and measuring detection performance is only possible up to a bounding-box.
[37] is currently the state-of-the-art for pixel-wise detection on the ETHZ-Shape dataset. This dataset contains 5 object categories and 255 images. We follow the experimental settings in [90]. The image set is evenly split into training and testing sets and performance is averaged over 5 random splits. Following [106] and [37], we report pixel-wise average precision (AP) on each class. The PASCAL criterion is used to decide if a detection is correct. The ground-truth segmentation masks were provided by [106].
Our results are displayed in table 4.1. Our method outperforms the state-of-the-art approach of [37] by 7% mean AP and our detection rate is comparable with the detection rate at 0.02, 0.3 and 0.4 FPPI in [37] (see table 4.2).

For the sake of completeness, we also evaluate our model on the level of bounding-boxes for the detected objects (standard setting). We used the INRIA horses dataset, which contains 340 images. Half of the images contain one or more horses and the rest are negative images. 50 horse images and 50 negative images are used for training. The remaining 120 horse images plus 120 negative images are used for testing. Results are listed in figure 4.5. Compared to [80], we improve the state-of-the-art detection rate at 0.1 fppi by 3.5% achieving a gain of 29% compared to the recent segmentation-based approach of [233].

Next, we evaluate the impact of our bottom-up grouping (see section 4.6) during training. For this experiment, the union of the first n best-ranked CMPC segmentation masks of [38] lying within the bounding-box were taken to define the bags (4.25). This setting

Table 4.2: Detection rate at 0.02, 0.3 and 0.4, fppi on ETHZ-Shape. We reach comparable pixel-wise detection rates to [37].

	Our Method	Carreira etal. [37]	Gu etal. [106]	Felz. etal.[80]
Apples	0.985/0.985/0.985	0.904/0.941/0.941	0.697/0.854/0.916	0.956/0.989/0.989
Bottles	0.860/0.975/0.975	0.891/0.975/0.975	0.745/0.932/0.958	0.835/0.981/0.981
Giraffes	0.830/0.924/0.924	0.920/0.970/0.970	0.543/0.736/0.800	0.675/0.936/0.943
Mugs	0.896/0.956/0.956	0.812/0.925/0.925	0.496/0.816/0.833	0.816/0.932/0.937
Swans	0.934/1/1	0.983/1/1	0.569/0.800/0.800	0.835/0.919/0.919
Mean	0.901/**0.968**/**0.968**	**0.902**/0.963/0.963	0.594/0.829/0.861	0.824/0.951/0.954

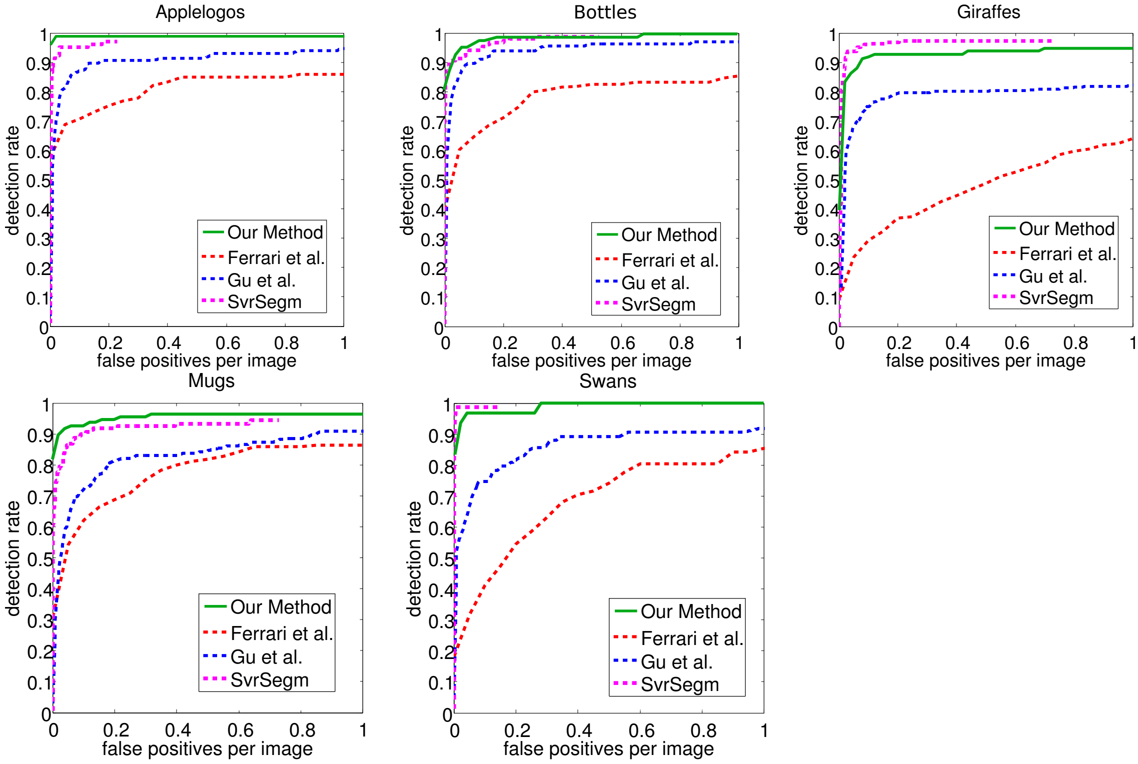


Figure 4.4: Detection Results on ETHZ-Shape classes. Our method outperforms state-of-the-art by 7% mean AP reaching a comparable detection rate at 0.02,0.3 and 0.4 FPPI

Method	AP	Det.rate at 0.1 FPPI
Our Method	**0.883**	**0.902**
[266]	-	0.730
BoSS [233]	-	0.630
[165]	-	0.652
[89]	-	0.674
[80]	0.871	0.867

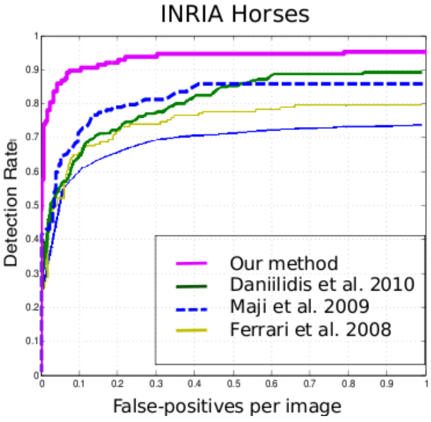


Figure 4.5: Detection results for the INRIA horses dataset. We improve [80] by 3.5% and the segmentation-based approach [233] by 29.9% detection rate at 0.1 FPPI.

would be equivalent to [37], which assumes that the best bottom-up generated segment covers the whole object. We varied the number of segments n and measured the detection performance in terms of average precision (AP). The experiment was evaluated on the horse category of PASCAL VOC 2007. The result is plotted on the left side of figure (4.6). For large n the performance reaches that of [80], since eventually all cells of m_0^j are active. Conversely, performance significantly drops as we approach n=1, which is the setting of [37]. Our full model is plotted as a constant line, since it is independent of the number of segments generated by [38].

In a second experiment, we tested the impact of our bottom-up grouping during testing. Instead of obtaining a bottom-up grouping for each sliding window, we tested our model exclusively on all the CMPC segments. We considered the tight bounding-box around each figure-ground segment S_t^I for an image I and used this segment to construct the bags B_i^j (in this case we have as many bags as segments S_t^I, see Eq. (4.25)). The experiment was carried out using the car category of VOC 2007 (see right plot in figure 4.6). We observed a 7.3% performance drop in AP. Hence, it is advisable to combine the different segments S_t^I (as we do) to obtain a better detection performance.

We also tested the impact of using a prototypical set of segments (see section 4.7) to represent an object shape. Since the matching score (4.26) uses a prototypical set of segments to evaluate each $d_i(.,.)$, we trained in this experiment a linear SVM using the Euclidean dot product (instead of using $d_i(.,.)$) between the feature representations $\phi_i(p_i^j, m_i^j)$ for all parts. In this case the matching score (4.26) is transformed into

$$\hat{d}(\mathcal{O}^j, \mathcal{O}^u) := \sum_{i=0}^{8} < \phi_i(p_i^j, m_i^j), \phi_i(p_i^u, m_i^u) > + < p_i^j - p_0^j, p_i^u - p_0^u > . \qquad (4.27)$$

In doing so, we obtained a very poor performance of 0.45 AP for the horse category compared to the 0.578 AP of our model.

To the best of our knowledge, there is no approach which explicitly tries to infer the overall object form using a model exclusively learned from bounding-box annotated training data for any category in the PASCAL dataset. In order to compare our approach with other detection methods we evaluate our model using the standard setting on the PASCAL VOC 2007 categories, where [80] best performs. In table 4.3 and figure 4.7, we observe that our model exhibits robust performance (43.68 MAP or Mean Average Precision)

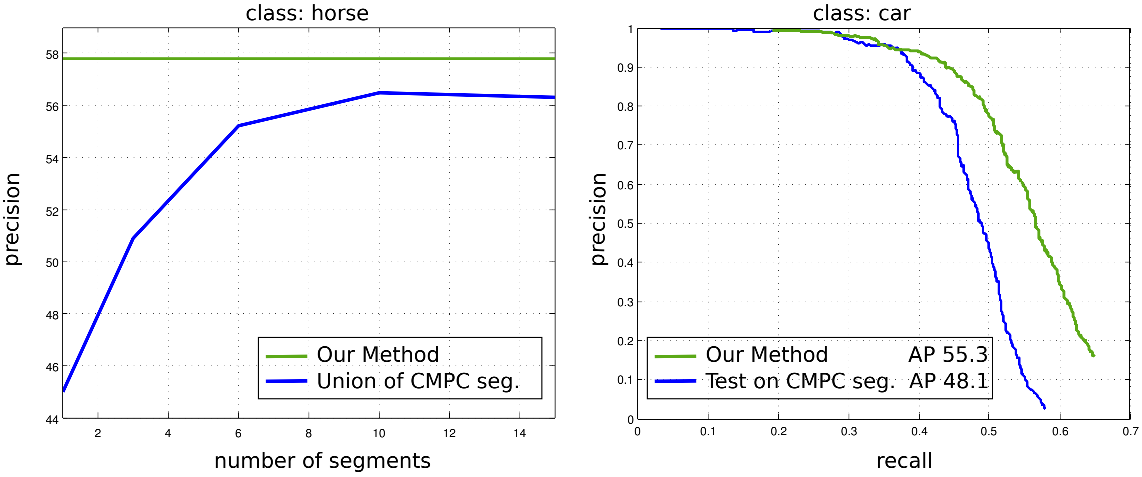


Figure 4.6: Impact of bottom-up grouping. Left: We trained our model using the union of the n best-ranked CMPC segments. Right: Test exclusively on CMPC segments.

under challenging image conditions at the same time that we obtain a richer output than just a bounding-box for detection. While [80] (42.34 MAP) is considered as our baseline model, we also listed comparable state-of-the-art detection methods. Due to the lack of exact precision numbers, the multi-feature approach of [243] is not listed in table 4.3. However, from the diagram presented in their paper, we read an approximate MAP of 42 for this set of categories and of 40 if [80] is evaluated exclusively on the proposed windows. Regardless of this, the strength of [243] remains in the usage of 5 different color features to train a Bag-Of-Words model. While we use a single, standard feature type, multi-feature approaches (e.g. [243, 245, 110]) are complementary and should enable further performance improvements.

4.10 Discussion

We have presented a model that explicitly represents object shape and segregates it from the background, *without*, however, requiring segmented training samples. The basis of this method is to capture the overall object form by grouping foreground regions in a model-driven manner and representing it through a class-specific prototypical set of segments automatically learned from unsegmented training data. By using exclusively bounding-box annotated training data, our model improves pixel-wise detection results and at the same time it provides a richer object parametrization for detecting object instances.
Furthermore, this model supports the thesis described in section 1.5.1 that increasing the description granularity at each stage of an object detection system leads to better results. In our case, this fine-grained description is introduce by directly using a richer object description which uses shape information directly during detection.

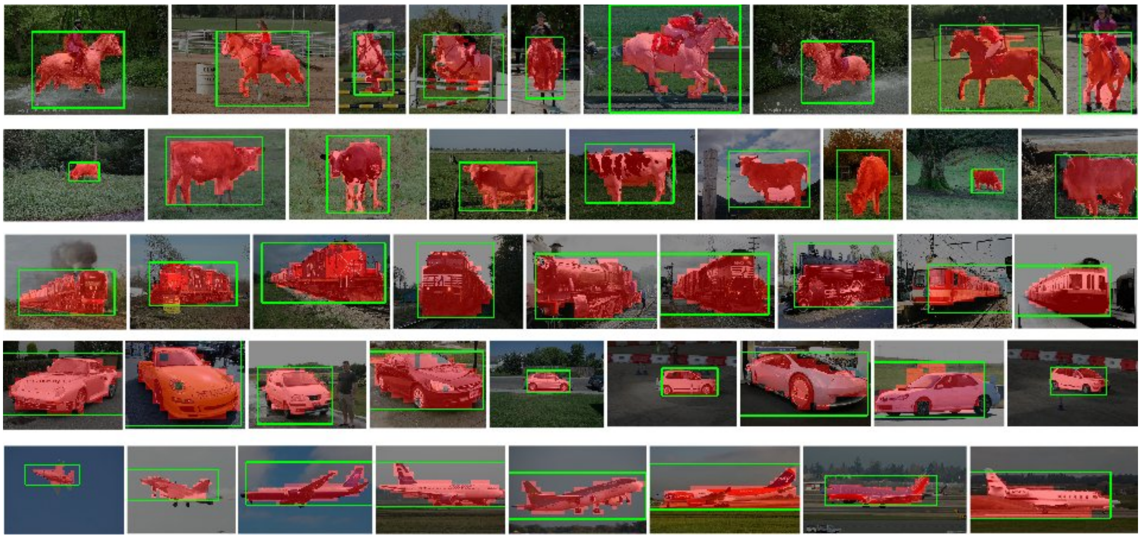


Figure 4.7: Detection examples for certain PASCAL VOC 2007 categories. The cells corresponding to the object foreground are grouped and used for detection.

Table 4.3: AP for best performing categories of [80] in PASCAL VOC 2007

	horse	cow	cat	train	plane	car	mbike	bus	tv	bicycle	sofa	person
Our approach	**57.8**	25.3	23.9	**47.8**	31.9	**59.8**	**49.8**	**51.6**	41.9	**59.8**	**33.7**	**41.9**
Felz. etal. [80]	56.8	25.2	19.3	45.1	28.9	57.9	48.7	49.6	41.6	59.5	33.6	**41.9**
best2007 [167]	37.5	14.0	24.0	33.4	26.2	43.2	37.5	39.3	28.9	40.9	14.7	22.1
UCI [65]	45.0	17.7	12.4	34.2	28.8	48.7	39.4	38.7	35.4	56.2	20.1	35.5
LHS [273]	50.4	19.3	21.3	36.8	29.4	51.3	38.4	44.0	39.3	55.8	25.1	36.6
C2F [198]	52.0	22.0	14.6	35.3	27.7	47.3	42.0	44.2	31.1	54.0	18.8	26.8
SMC [207]	51.0	23.0	16.0	41.0	26.0	50.0	45.0	47.0	38.0	56.0	29.0	37.0
HStruct [220]	48.5	18.3	15.2	34.1	31.7	48.0	38.9	41.3	39.8	56.3	18.8	35.8
LatentCRF [221]	49.1	18.5	14.5	34.3	31.9	49.3	41.9	49.8	41.3	57.0	23.3	35.7
MKL [245]	51.2	**33.0**	**30.0**	45.3	**37.6**	50.6	45.5	50.7	**48.5**	47.8	28.5	23.3

CHAPTER 5

BEYOND ANNOTATED DATASETS - PARAMETRIC OBJECT DETECTION FOR ICONOGRAPHIC ANALYSIS USING SHAPE EQUIVALENCE

In chapter 1.4 we wrote about the importance of shape for both the human visual system and for computer vision. One of the capabilities of the human visual system regarding shape is the ability of detecting *shape equivalence* which refers to the ability of distinguishing two *different* objects as having the same shape. The present chapter introduces a method that exemplifies how this property can be used for solving specific computer vision tasks. The idea in this chapter consists of indirectly detecting important objects within an image by finding reoccurring patterns which share the same shape. Specifically, the present chapter develops a computational method for *unsupervised* object detection for use within the field of cultural heritage and shows how fruitful the interaction between computer vision and cultural heritage is. The results presented in this chapter were first published in [183].

5.1 Parametric Object Detection for Iconographic Analysis

Every iconographic analysis within the field of cultural heritage needs to begin with what can be seen in the objects being considered. Based on these observations the objects under analysis are compared with other visual images. For this purpose research is needed to understand how a particular example differs from others and why these differences matter. Furthermore, the co-occurrence of similar patterns within the image corpus also reveals important common characteristics and relevant structures, which in turn are used to infer meta-information (e.g. the artistic choice of a group of artists) involved in the process of the image generation. An iconographic analysis may involve intensive screening of thousands of visual images in order to establish a consistent interpretation. Therefore, in such cases the use of automatic object detection systems is required. However, current state-of-the-art object detection systems rely on one key aspect that is not always fulfilled in iconographic analysis tasks: Similar to the human learning procedure, computer systems require training examples in order to learn a model for object instances that are to be searched in new images. For the task we explore in this chapter we lack any such training data, i.e., we need to search objects that we have not seen before.

Specifically, in this chapter we analyze images taken from Chinese comics digitized at the Cluster of Excellence “Asia and Europe”. The focus of the digitization process is on comics from the second half of the Cultural Revolution and immediately thereafter, which was the heyday of comic production in the ”small people’s books” (xiaorenshu) format. In these books a special type of emphasis is used to accentuate heroes, objects, or idols like the image of Mao Zedong: they are depicted as a sun omitting rays of light. This is an example of how the occurrence of similar shape patterns in different images reveals a meta-information (i.e. the accentuation of an object) related to the intention of the artists which drew the image. Automatically finding the accentuated objects is a preliminary step to carry out an iconographic analysis, which may reveal the intention of the cartoonist, or more importantly the intention of those who commissioned those comics. Therefore, to detect emphasized objects may help to reveal possible programmatic shifts in the focus of the stories.

In the present chapter we develop a novel system which automatically finds emphasized objects by detecting co-occurrent similar shape patterns (in this case the irradiation of the object) in the image corpus. This task is non-trivial since finding the objects of interest requires finding the rays which surround them. However, recognizing which line segments in the image belong to a ray annulus and which do not requires knowing where the objects are localized. Furthermore no training information is available, which makes it impossible to directly search for the object as would be the setting scenario in object detection systems [80, 146, 266].

Image databases in the field of cultural heritage are normally made accessible via textual annotations [7], or more recently [265] presented an annotated dataset for purposes of object detection. The problem of analyzing unlabeled datasets within this field remains an unsolved problem. Object detection and recognition is a widely studied research area within computer vision as we saw in chapter 2. However, the main work in object detection concentrates either on a supervised or semi-supervised learning (e.g. [80, 146]), which relies on annotated training data. In the present chapter we intend to detect objects *without* any training information. Finally, our work is related to different tasks within the field of computer vision, such as contour detection [125, 171, 35, 163, 173], clustering [74, 128, 122], and Hough Voting [117, 105, 165, 98].

5.2 Using Shape Equivalence for Object Detection

As stated in the introduction, we develop a system, which automatically finds objects in an unsupervised manner. The class of objects we are interested in is characterized by a surrounding circle of fragmented light rays. The idea is to consider these annuli of light rays as shape equivalent objects (i.e. objects that share a similar shape) and use their detection to solve the original problem (i.e. detecting the enclosed objects). Therefore, we use the observation that circular line patterns surrounding the object may intersect at one point or at least, due to noise, in a region with high density of line intersection points (see Fig. 5.1 for an overview of our method).

To describe the system, we subdivide the method in 3 steps: Edge Extraction, Line Fitting and Clustering, and Object Localization and Detection. In the following, we describe the different steps of our method.

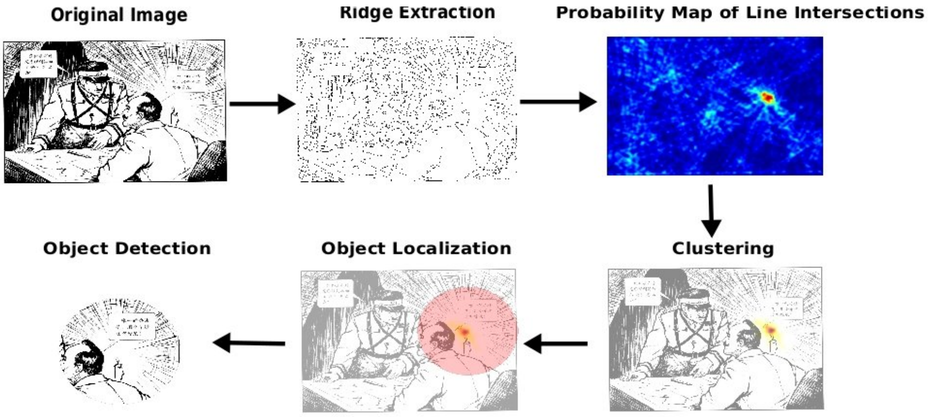


Figure 5.1: Overview of our method. This example automatically finds the object in the image, which was emphasized by the author of the comic.

5.3 Edge Extraction

As a preprocessing step in our method, we firstly extract edges from this kind of images. Common edge extraction methods like Canny [35] or Pb [173] fail in accurately extracting edges in this type of images. This is the case since Canny uses a filter which cannot handle lines of varying thickness. On the other hand, Pb uses different cues like color and texture for the edge extraction. Both cues are not available in any image of the database we are considering.

To avoid the drawbacks presented by these methods, we firstly convolve the image with different Laplace of Gaussian (LoG) Filters of varying sigmas. The use of this kind of filter is suitable since it allows obtaining a single response for lines of varying thickness. In our experiments we use the sigma values $\sigma = 0.8 + j * 0.4 \; j = 1, \cdots, 9$. For every pixel in the image we then take the maximal response over all sigmas. This ensures in praxis a good contrast between ridge response and background. Finally, non-maximum suppression followed by hysteresis thresholding is applied to obtain the final edges in the image. In Figure 5.2 (a)-(b) we observe an example of how common methods fail to extract edges, while our method (c) is able to cope with the difficulties in the images we are considering here.

5.4 Line Fitting and Clustering

In the following, we introduce an important technique in computer vision which builds a framework for our method.

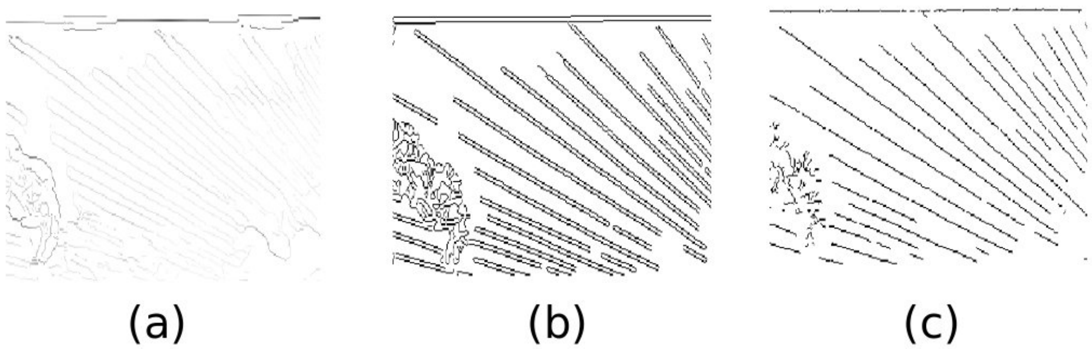


Figure 5.2: Edge Extraction. (a) Pb Edges. (b) Canny Edges. (c) Our method.

Hough Transform

The Hough Transform is a technique for finding instances of objects sharing certain shapes by a voting procedure. It was first introduced by Richard Duda and Peter Hart in 1972 [73] for detecting lines and circles and Ballard extended it for arbitrary shapes [8].
The simplest case of Hough Transform is used for detecting straight lines. The main idea is to consider the lines not as image points but instead in terms of parameters. Normally, due computational accuracy, instead of representing a line using the slope and the offset $y = mx + b$, a polar-coordinates representation is used. In this representation, whereas the parameter r represents the distance between the line and the origin, the angle θ is the angle of the vector from the origin to this closest point resulting in

$$y = \left(-\frac{\cos\theta}{\sin\theta}\right)x + \left(\frac{r}{\sin\theta}\right). \tag{5.1}$$

Using this representation, each line in the image can then be identified with a pair (r, θ), $r \in \mathbb{R}$, $\theta \in [0, \pi)$. The (r, θ)-plane is referred to as *Hough space* or *Hough accumulator*. In this particular space, a line corresponds to a sinusoidal curve and if the curves, corresponding to two points are superimposed, the location (in the Hough space) where they cross corresponds to a line (in the original image space) that passes through both points. Thus, the problem of finding collinear points transforms into the problem of finding intersecting curves.

As stated in the introduction, objects of interest are surrounded by a circle or semi-circle of fragmented light rays. This means that object centers are characterized as the intersection of many lines or at least, due to noise, as a region with high density of line intersection points. For this reason, we first need to detect all straight lines which appear in the drawing. This is done in an iterative manner:
Firstly, a pixel is selected at random and then a line is hypothesized by using its own and the neighboring pixel's gradient orientation. This line is then extended (pixel-wise) by grouping pixels with similar orientation in the direction of the hypothesized line. If the circular variance of gradient orientations on the fitted line exceeds a predefined threshold, the line-growing process breaks and the algorithm starts from the beginning. After this procedure ends, all pixels used to calculate the line are removed from the search list. The algorithm finishes when the search list is empty. As a post-processing step, all lines which

do not exceed a minimal length are removed.

Since we are interested in detecting high-density regions of line intersections, we calculate as a further step in our method all possible intersections between the fitted lines by the iterative algorithm. For this, we construct a Hough Accumulator using polar coordinates for every fitted line. Each line is then weighted by its circular variance. This allows us to decrease the importance of lines which are not completely straight and therefore do not belong with a high probability to any ray pattern. From the Hough Accumulator we then extract all possible intersections. In our experiments we obtain in this manner around 100000 line intersections per image.

To localize high-density line intersection regions, we then cluster all intersections obtained from the Hough Accumulator using a hierarchical clustering (Single-Linkage). Due to noise in the line fitting algorithm, every intersection x_{ij} of two lines l_i, l_j renders an uncertainty, which we model using a 2D Gaussian distribution centered at the intersection point x_{ij} and with a covariance matrix determined by the angles of the two lines which intersect. Specifically, we have $\mu_{ij} = x_{ij}$ and the covariance matrix is defined as

$$\Sigma_{ij} = \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix}, \tag{5.2}$$

where $\sigma_1 = \tan(\theta_1/2)$ and $\sigma_2 = \tan(\theta_2/2)$. Here, θ_1 is the smallest and θ_2 the greatest angle of the two intersecting lines. For orthogonal lines, our model yields for Σ_{ij} the identity matrix $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

Using this uncertainty model, the pairwise distance measure for the hierarchical clustering is defined in a probabilistic manner. For two intersection points x_{ij}, x_{mn} we define:

$$d(x_{ij}, x_{mn}) := p(x_{ij} \mid \mu_{mn}, \Sigma_{mn}) * p(x_{mn} \mid \mu_{ij}, \Sigma_{ij}), \tag{5.3}$$

where $p(x \mid \cdot\, \mu, \Sigma)$ is the probability of x, conditioned on the model $\{\mu, \Sigma\}$. This clustering procedure using the measure (5.3) results in the desired object center hypothesis.

Finally, it is interesting to remark that our measure (5.3) is related to the Mahalanobis distance, but our distance is more flexible since it allows for rotation of the Gaussian distributions.

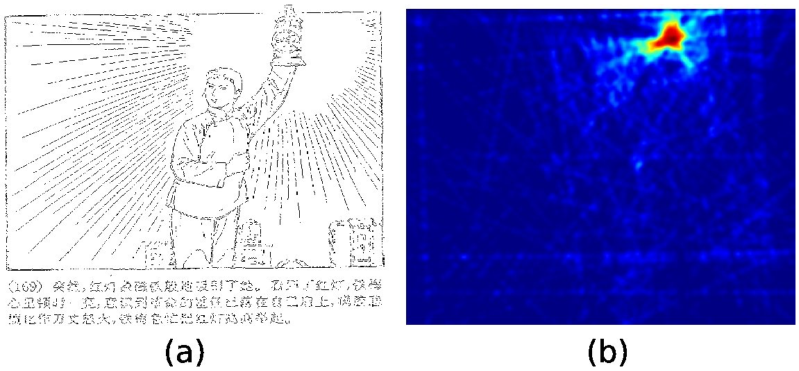

（169）突然，红灯在磁铁般地吸引了他。看见了红灯，铁梅心里顿时一亮，意识到革命的道路已落在自己肩上，满腔悲愤化作万丈怒火，铁梅拿起红灯高高举起。
(a)
(b)

Figure 5.3: Object Center estimation using a Gaussian model. (a) Ridge Image. (b) Line Intersections.

5.5 Object Localization and Detection

In the last section, we obtained all hypotheses of relevant objects of interest. Since the object center does not yield any information about the size of the object itself, we calculate its scale using the information contained in the annulus of light rays surrounding the object center. More specifically, the scale of the object is determined as the radius from a circle centered at the object center hypothesis (described in the last section) to the beginning of the circular ray pattern. The beginning of this ray pattern can also be characterized as a steep increase in the line density. Using this observation, we first weight all straight lines according to how close they are to the object center hypothesis: lines crossing the center or passing nearby should get a high weight since they belong to the rays describing the object center. All other lines should get a small weight and should not play any role in the line density estimation. Once we weight all line pixels in the image according to the object center hypothesis we calculate the density as the sum of the weighted pixels within a certain radius of a circle centered at the object center. Specifically, given the object center x_c and the set of indices $I_r := \{i \,|\, |x_r - x_c| <= r\}$, the density function is given by

$$\text{density}(r) = \sum_{i \in I_r} w_i, \tag{5.4}$$

where w_i is the corresponding weight of the pixel x_i. Given this density function, to find the radius from where the circular ray pattern starts, we calculate the maximum of the first derivative of the density function. In practice, we also calculate the maximum of the first derivative of the radius-normalized function of the cumulative circular variance to improve the location of the radius.

An example of this scale estimation procedure for an object center can be seen in Fig. 5.4. Fig. 5.4 (a) shows how the object can be parametrized by a circle, and Fig. 5.4 (b) shows the density function. Its first derivative is shown in (c). The maximum of the first derivative is marked with a green point. The red point in (e) shows the improved length of the radius which corresponds to the red circle in (a).

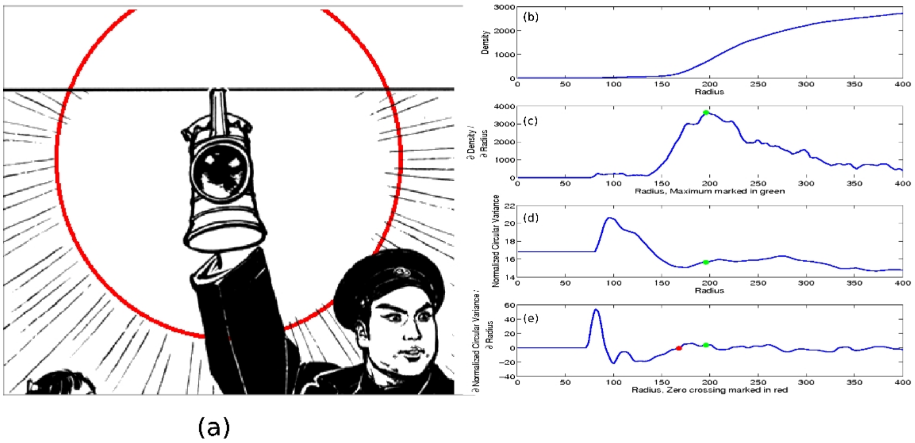


Figure 5.4: Scale Estimation. (a) Circle-parametrized scale estimation. (a)-(b) Density Function and First derivative. (c)-(d) Normalized Circular Variance and First Derivative

5.6 Results

The Cluster of Excellence “Asia and Europe” developed the Chinese Comic Database. As stated in the introduction, this database focuses on comics from the second half of the Cultural Revolution and immediately thereafter. Most of the database consists of black and white cartoon drawings. In figure 5.5 we can see some results generated by our method. The first column shows the original image presented to the system. In the second column we present results of the ridge extraction procedure. For purposes of visualization we show only a section of each image. After the line fitting process we calculate all line intersections using our probabilistic model, this is shown in the third column of figure 5.5. The last column of the figure shows the object localization calculated by our system, as described in the last section.
In the 5th row of figure 5.5 we can see how our system successfully extracts the object center of the light rays. Specifically, if we see the line-intersection map we can see how the map-energy within the circle has 3 centers: the person, the lamp and the sun. This means that the light-rays intersect in three different centers, thus it is clear how the artist of this comic draws the light rays to emphasize different objects in the image. Further, in the 4th row of figure 5.5 we can clearly see how our method correctly extracts the scale of the object, fitting the circle in such a way that the whole object of interest is covered by it.

5.7 Discussion

In this chapter we have presented a novel method which enables to automatically find the irradiating objects within a corpus of images taken from the Chinese Comics Database digitized from the Cluster of Excellence “Asia and Europe”, which focuses on comics drawn during the second half of the Cultural Revolution and immediately thereafter. This tool will enable researchers to screen large amounts of data, visualize relevant objects and carry out an exhaustive iconographic analysis of the whole database. Furthermore, we hope this approach will help to show new ways of interdisciplinary research and mutual benefits between cultural heritage and computer science.

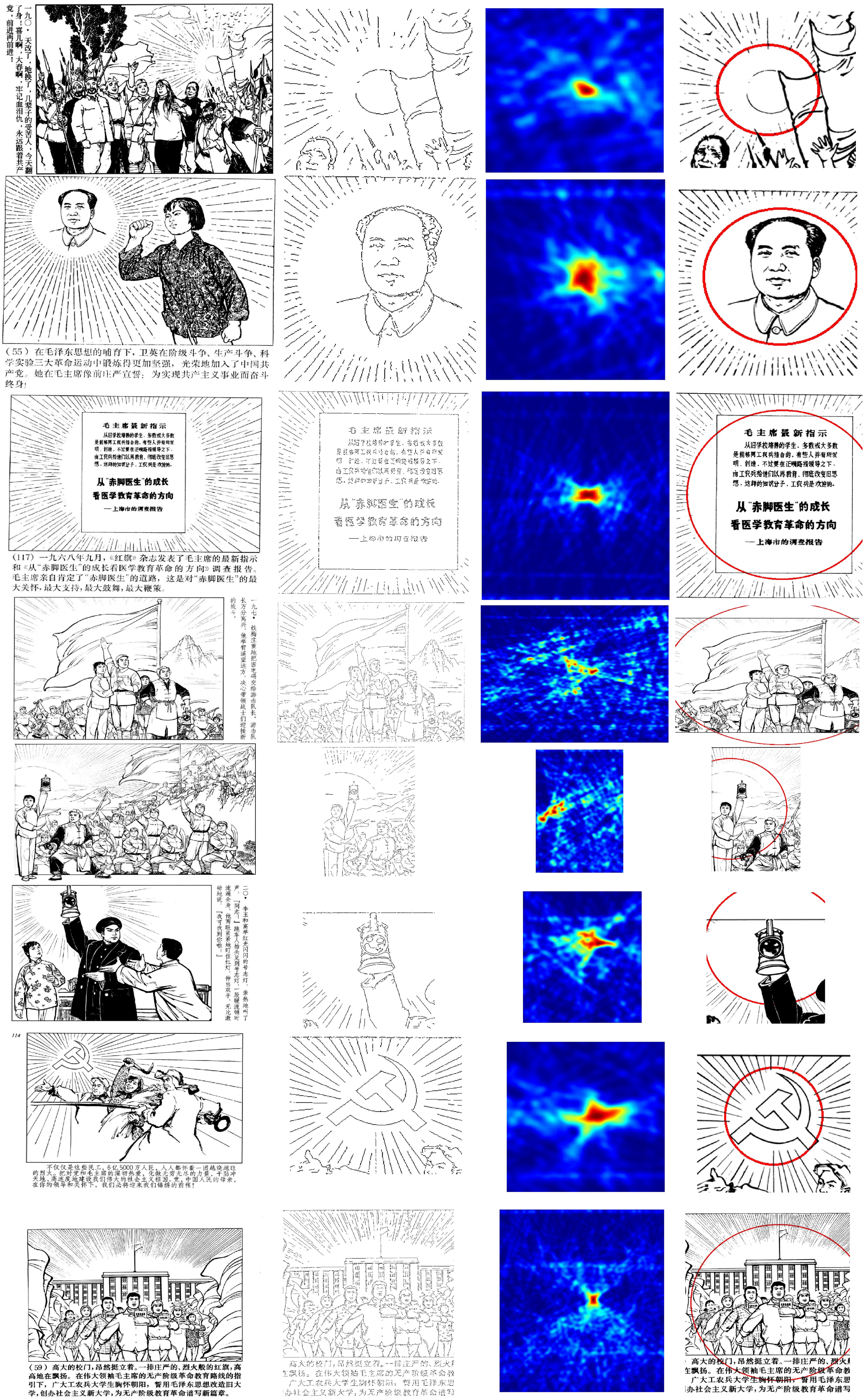

一九〇・天改了，地换了，几辈子的受苦人，今天翻了身！喜儿啊，大春啊，牢记血泪仇，永远跟着共产党，前进再前进！
（55）在毛泽东思想的哺育下，卫英在阶级斗争、生产斗争、科学实验三大革命运动中锻炼得更加坚强，光荣地加入了中国共产党。她在毛主席像前庄严宣誓：为实现共产主义事业而奋斗终身！
毛主席最新指示
从旧学校培养的学生，多数或大多数是能够同工农兵结合的，有些人并有所发明、创造，不过要在正确路线领导之下，由工农兵给他们以再教育，彻底改变旧思想，这样的知识分子，工农兵是欢迎的。
从"赤脚医生"的成长
看医学教育革命的方向
—上海市的调查报告
(117) 一九六八年九月，《红旗》杂志发表了毛主席的最新指示和《从"赤脚医生"的成长看医学教育革命的方向》调查报告。毛主席亲自肯定了"赤脚医生"的道路，这是对"赤脚医生"的最大关怀，最大支持，最大鼓舞，最大鞭策。
不仅仅是这些民工，6亿5000万人民，人人都怀着一团越烧越旺的烈火。把对党和毛主席的深切热爱，化做无穷无尽的力量，干劲冲天地，高速度地建设我们伟大的社会主义祖国。党，中国人民的母亲，在你的领导和关怀下，我们必将迎来我们锦绣的前程！
（59）高大的校门，昂然挺立着。一排庄严的、烈火般的红旗，高高地在飘扬。在伟大领袖毛主席的无产阶级革命教育路线的指引下，广大工农兵大学生胸怀朝阳，誓用毛泽东思想改造旧大学，创办社会主义新大学，为无产阶级教育革命谱写新篇章。

Figure 5.5: A small subset of the results provided by our method.

CHAPTER 6

DESCRIBING SHAPE CHANGES

As described in section 1.5.2 computer vision is impelled to go beyond the mere fact of searching for objects and describing them by means of bounding-boxes. Moreover, the object itself must be understood and therefore it is crucial to analyze its structure and shape. It is an important underlying idea of this thesis that the analysis of the structure and morphology of an object can be carried out by analyzing the deformation of the shape between two distinct but similar objects.

The present chapter aims at introducing representative approaches within the field of computer vision that have been developed for describing shape changes. The study of such previous models will reveal the important model paradigms of current state-of-the-art approaches and at the same time this study will also show important limitations of current approaches. Based on this analysis, in the next chapter we will introduce a novel method capable of tackling the full problem introduced in section 1.5.2.

6.1 Shape Changes Under Given Correspondences

An early technique for describing shape similarity in two dimensions is *Procrustes analysis*, a term which was introduced in factor analysis by the authors in [119]. In its simplest version (the generalized version over more dimensions is similar, but our interest is focused on 2D images), this methodology represents a pair of shapes by a centered set of corresponding complex *landmark points*, that is $y = (y_1, \cdots, y_k)^T$ and $w = (w_1, \cdots, w_k)^T$ both in $\mathbb{C}^k$ such that $w^* \mathbb{1}_{[k]} = 0$ and $y^* \mathbb{1}_{[k]} = 0$, where y^* denotes the transpose of the complex conjugate of y. Two main steps can be identified in the Procrustes Analysis. Firstly, both shapes w and y are matched using similarity transformations (called *full Procrustes fit*) and secondly, the distance between shapes called *full Procrustes distance* is defined. The full Procrustes fit is a standard least squares solution with complex variables for scale, rotation, and translation. These are estimated by minimizing

$$\min_{\beta,\theta,a,b} \|q - w\beta e^{i\theta} - (a+ib)\mathbb{1}_{[k]}\|^2 \tag{6.1}$$

yielding the solution (s. [70])

$$a + ib = 0 \tag{6.2}$$

$$\theta = arg(w^*y) \tag{6.3}$$

$$\beta = \frac{(w^*yy^*w)^{1/2}}{w^*w} \tag{6.4}$$

In case that the landmark points are normalized to have unit size, that is

$$\sqrt{y^*y} =, \sqrt{w^*w} = 1. \tag{6.5}$$

the full Procrustes distance between the shapes is given by

$$d_F(w, y) = \inf_{\beta,\theta,a,b} \| \frac{y}{\|y\|} - \frac{w}{\|w\|}\beta e^{i\theta} - a - ib\| \tag{6.6}$$

$$= \{1 - \frac{y^*ww^*y}{w^*wy^*y}\}^{\frac{1}{2}}. \tag{6.7}$$

Within the Procrustes analysis, the mean shape of a population or the variability in shape can also be described using this distance ([49, 131]).

A major problem of measuring shape distances using the full Procrustes distance (and all its generalizations described e.g. in [70]) is the fact that we obtain a numerical value for shape comparison but it does not indicate locally where the objects differ and what nature the differences have.

Such a study of localized differences between shapes can be traced back to D'Arcy Thompson's [231] transformation grids. In his book D'Arcy Thompson manually placed rectangular squared grids (called Cartesian grids) on both shapes and considered the transformations of different grid blocks between corresponding biological parts, enabling him to describe the *shape change* between two species. Thompson's idea was that these comparisons would reveal the origins of form. Similar approaches to D'Arcy Thompson can even be found earlier in history within the field of cultural heritage, specifically with Renaissance artists of the 16th century. For instance, A. Dürer (1528) [75] used affine transformed grids of human bodies and their parts with the finality of exploring human body proportions and to study the limits of normal variation in shape.

The simplest possible manner to describe the changes in shape and size using Cartesian grids would be to use a global affine transformation. All grid blocks would uniformly deform, and parallel lines in the first shape would remain parallel after applying the transformation. In fact, some examples of affine-transformed grids were already described by D'Arcy Thompson. However, it is clear that only very limited shape deformations can be described using a single global affine transformation. A more complicated transformation was introduced by Sneath [227], who approximated the grid using cubic polynomials and representing each of them by interpolating coefficients. However, it was not until 1989 when Bookstein [27], borrowing ideas from the mathematician Duchon and Meinguet [72, 177], developed a successful approach for corresponding landmark points using a pair of thin-plate splines.

Thin-plate Splines

In this approach k landmark points $x_j \in \mathbb{R}^{2\times 1}$, $j = 1, \cdots, k$ on the first shape are mapped to exactly k points y_i using different interpolation-functions for each coordinate

$$x_{ir} = \phi_r(y_j),\ r = 1, 2,\ j = 1, \cdots, k \tag{6.8}$$

resulting in a bivariate transformation function $\phi(y_j) = (\phi_1(y_j), \phi_2(y_j))$

$$\phi(z) = z \cdot d + \sum_{i=1}^{K} \psi(\|z - y_i\|) \cdot c_i, \tag{6.9}$$

where all points are represented using homogeneous coordinates, that is, each point z_i is represented as a vector $(1, z_{i1}, z_{i2})$. Furthermore, $d \in \mathbb{R}^{3\times 3}, c \in \mathbb{R}^{K\times 3}$ and $\psi(\|z - y_i\|)$ is a radial basis function (RBF). The function (6.8) is known as a pair of thin-plate splines (PTPS) [70][1]. It can be shown (e.g. [250]), that the parameters c, d in Eq. (6.9) can be found minimizing the energy function

$$\begin{aligned} E_{TPS} &= \min_{d,c} \|X - Yd - \Psi c\|^2 + \lambda \sum_{j=1}^{2} \int \int_{\mathbb{R}^2} \left(\frac{\partial^2 \phi_j}{\partial x^2}\right) + \left(\frac{\partial^2 \phi_j}{\partial x \partial y}\right) + \left(\frac{\partial^2 \phi_j}{\partial y^2}\right) dxdy, \\ &= \min_{d,c} \|X - Yd - \Psi c\|^2 + \lambda Tr(c^T \Psi c) \end{aligned} \tag{6.10}$$

where Y, X are the concatenated versions of the points y_i, x_i and Ψ is a $k \times k$ matrix formed with the entries $\psi(\|y_j - y_i\|)$ (also called TPS kernel). The second term in 6.10 is called *bending energy* and receives its name from a physical analogy involving the bending of a thin sheet of metal, where the deformation occurs in the z direction orthogonal to the plane [250]. This approach based on thin-plate splines (TPS) consists of an affine part (matrix d) and a non-affine component (matrix c), which parametrizes the non-affine deformation for every point. The QR-decomposition of

$$Y = \begin{pmatrix} Q_1 & Q2 \end{pmatrix} \begin{pmatrix} R \\ 0 \end{pmatrix} \tag{6.11}$$

can be used to estimate the exact minimizers c, d in a closed-form solution [250]:

$$\gamma = (Q_2^T \Psi Q_2 + I_{K-3})^{-1} Q_2^T X \tag{6.12}$$

$$d = R^{-1} Q_1^T (X - \Psi Q_2 \gamma) \tag{6.13}$$

$$c = Q_2 \gamma \tag{6.14}$$

The TPS can be considered as a special case of *Kriging* [134] for 2 dimensions. Therefore, this approach can be extended to non-planar more-dimensional shapes using Kriging interpolation methods (e.g. universal and intrinsic Kriging or Kriging with derivate constraints)[70].

[1] The name pair of thin-plate splines refers to the fact that ϕ is a bivariate function with a Thin Plate Spline in each component

Finite Element Analysis

Another possible method for describing shape changes is to use the variational approach of *Finite Elements Method* (FEM), which uses a similar idea to the Cartesian grids of D'Arcy (in cases of linear Finite Elements). This method has been explored e.g. in [25, 186, 132, 152, 42]. The FEM method was originally developed for finding approximate solutions to boundary value problems. The application of this method for describing shape changes works by first generating a grid or mesh over the shape y which is going to be transformed to another shape x. Each block of this mesh is referred to as a *finite element* and can have different forms. For 2D spaces either triangles or quadrilaterals can be used. Finite elements may contain a different amount of landmarks but in the simplest case the mesh is chosen such that each landmark point of the shape lies on the vertices of the element. Thereafter, within each finite element, a set of n piecewise functions are defined which separately interpolate the coordinates of the shape x using the points of y. The interpolation function is uniquely defined using boundary conditions relative to the neighboring elements which guarantee a continuous interpolation between finite elements. Depending on the polynomial degree of each interpolant function different numbers of boundary conditions are required to be defined. In the simplest case of triangles arising from a Delauny triangulation (e.g. [192]) the interpolation within each finite element results in an affine transformation and thus, the global function describing the shape change consists of a piecewise affine transformation. This linear transformation model is essentially equivalent to the recently introduced method in [116], which also defines affine transformations over a Delauny triangulation with boundary constraints over the different triangles.

Piecewise Affine Transformation Models

In the last section we saw how a piecewise affine transformation model arose as a special case of linear finite elements over an automatically generated mesh (using triangles as blocks in the grid). However, mesh-free piecewise affine models have also been introduced in literature. Some of these methods have been developed specifically for the registration of articulated structures (e.g. [59, 197]). Methodologies for interpolating between local transformations [199] as well as for assuring global invertible transformations [48, 4, 188] also exist. However, common to all these methods is that the affine-transformed local structures in the shape are manually chosen. That means that not only the correspondences between landmark points (or pixels) need to be known, but additional information is required: the spatial localization of local shape changes needs to be known a priori.

Piecewise Affine models have also been used in the context of *sparse motion segmentation* [253, 28]. The goal there is to decompose videos into similarly moving layers. Specifically, in [253] (the work [28] is a modification thereof) the scene is firstly divided into a regular grid and an affine transformation is calculated for each block. Thereafter, the affine parameters are clustered (using a modification of Kmeans) to reduce the number of components. In a second step the authors iterated between the assignment of points in the scene to the affine components based on its registration error (and Euclidean distance) and the refinement of the components themselves. Compared to other methods for linear subspace segmentation like [78, 139], where multiple frames are needed to correctly separate the different motions present in the sequence, Wang and Adelson [253] utilized only two frames at a time for the analysis.

Normally, algorithms defined for motion segmentation deal only with frames coming from the scene and thus, the task reduces to the motion analysis of the *same* objects through time (if we exclude appearances or disappearances of objects). For shape registration where we are also interested in describing the shape changes between *different* objects, we additionally have to deal with clutter, missing contours and an accurate estimation of small and continuous deviations in transformations. Thus, the task of using piecewise affine models for shape analysis becomes harder to solve.

6.2 Shape Change Under Missing Correspondences

An additional difficulty in describing changes between shapes arises when the correspondences between shapes (i.e. between pairs of landmark points) representing the shapes are missing. This situation is commonly found in real-life applications when similar shapes belonging to different objects are analyzed. In this case the problem of finding a transformation model which describes the shape changes and the correspondences between points is intrinsically related. In order to estimate the transformation model, correspondences between points or shapes need to be known. The difficulty of this scenario becomes clear by formulating the problem as

$$\min_{T,C} E(T,C) \tag{6.15}$$

where E is the energy term to be minimized, T is a given transformation model (e.g. an affine transformation or a TPS) and $C \in \mathbb{B}^{m \times n}$ is a binary matrix specifying the correspondences between m points in a shape Y to n points in another shape X. If the chosen transformation model is continuous then the optimization problem turns into a mixed-integer program with a very large optimization space. Already in order to estimate one-to-one correspondences there exist alone $\mathcal{O}(m^n)$ possibilities for defining correspondences between points in both shapes.

The solution for problem 6.15 has been approached from different points of view. In the following we will describe the most relevant approaches for solving this joint problem.

Iterative Closest Point Algorithms

Since its introduction [40, 19] for registering range-data imagery, the Iterative Closest Point (ICP) algorithm has experienced several modifications. In its original version, if shape Y consisting of a set of landmark points has to be registered to shape X, the ICP algorithm greedily minimizes the total registration error by alternating two steps:

- Estimation of closest point: For each point $y_i \in Y$, estimate its closest point in X, that is
$$\min_{x \in X} \|x - y_i\| \tag{6.16}$$
- Estimation of the transformation model: Use the correspondences obtained in the first step to calculate a global affine transformation T. Thereafter, shape Y is transformed using T

The iteration is stopped when the change in mean-square error (MSE) between the points in both shapes falls below a given threshold. It can be proved that this algorithm monotonically converges to a local minimum with respect to the mean-square distance objective function ([19]). This convergence is given since the reduction of the squared-error for each point also reduces the MSE for all points together.

The popularity of this algorithm can be seen from the numerous subsequent modifications it went through over the years (e.g. [236, 174, 22, 76, 205]). However all these approaches keep the same underlying idea of the algorithm modifying only special issues which can be classified ([214]) according to (a) selection of used points (b) selection of matching points (c) outliers selection or (d) error metric minimization.

Non-Rigid Point Matching

Inspired by the ICP, Chui and Rangarajan [44] introduced a new algorithm called RPM (i.e Robust Point Matching) for solving both problems: finding the correspondences between landmark points and estimating a transformation model. However, the first difference to the ICP algorithm consists in the usage of a TPS to model the deformation between shapes. Thus, this approach can be seen as an extension to the approach of [27], where given the correspondences between shapes the shape change was also modeled with a TPS. A further difference to the ICP algorithm is that Chui and Rangarajan cast the problem in a least-squares optimization algorithm allowing one-to-one *fuzzy correspondences* between points (in contrast to the nearest-neighbor heuristics of the ICP algorithm) and *deterministic annealing* (DA) is used to overcome local minima during the optimization process.
Given two shapes Y and X, with points $y_i, x_i \in \mathbb{R}^{2\times 1}$, the correspondence between points is given by a real-valued matrix $C \in \mathbb{R}^{m\times n}$, $c_{ia} \in [0, 1]$. To handle outliers the RPM algorithm [44] adds an extra column and row to the matrix C allowing for this extra *outlier point* many-to-one matching. The resulting optimization problem is given by

$$\min_{C,f} E(C,f) = \sum_{i=1}^{n+1}\sum_{a=1}^{m+1} c_{ai}\|y_i - f(x_a)\|^2 + \lambda T\|Lf\|^2 \tag{6.17}$$

$$+ \quad T\sum_{i=1}^{n+1}\sum_{a=1}^{m+1} c_{ai}\log c_{ai} + T_0 \sum_{a=1}^{m+1} c_{a,n+1}\log c_{a,n+1} \tag{6.18}$$

$$+ \quad T_0 \sum_{i=1}^{n+1} c_{m+1,i}\log c_{m+1,i} \tag{6.19}$$

$$\tag{6.20}$$

subject to the one-to-one matching constraints

$$\sum_{i=1}^{n+1} c_{ai} = 1,\ a = 1,\cdots,m \tag{6.21}$$

$$\sum_{a=1}^{m+1} c_{ai} = 1,\ i = 1,\cdots,n \tag{6.22}$$

The parameter T together with the $c\log c$ terms regulate the Deterministic Annealing schedule. Whereas high values of T allow fuzzy correspondences, lowering the parameter produces in the limit a binary matric C. In Eq. (6.17) the transformation f and the corresponding regularizer $\lambda T\|Lf\|^2$ are defined as in Eq. (6.10) where the TPS was introduced. Due to the complexity of minimizing (6.17) the RPM algorithm obtains only a local minimum (given initial correspondences) alternating between the estimation of the correspondence matrix C and the TPS estimation f. In order to enforce the constraints (6.21) the authors run the Sinkhorn algorithm at each iteration, which iterates row and column normalizations until convergence. However, it is important to remark that Sinkhorn's algorithm only guarantees convergence for squared matrices with values within $[0, 1]$. Therefore, a perfect fulfillment of the constraints (6.21) using a different number of points is not guaranteed. Despite the clear improvement of this algorithm w.r.t to the ICP algorithm in that a sound optimization algorithm is formulated, the RPM is very sensitive to the parameter setting of T (which determines DA scheduling) and the regularizer for the TPS calculation. Furthermore, the local structure of the shape is disregarded by ignoring pairwise relations between points during the estimation of C.

Coherent Point Drift Algorithm

The work of Myronenko et al. [187] (also called CPD algorithm) can be seen in the tradition of the ICP and the RPM algorithms described above. The CPD algorithm conceives the points in the shape $Y \in \mathbb{R}^{m\times 2}$ as the centroids of a Gaussian mixture model (GMM) and the points in the other shape $X \in \mathbb{R}^{N\times 3}$ as the data points generated by the GMM. Therefore, the authors consider the alignment of two point sets as a probability density estimation problem, where one point set represents the GMM centroids and the other one represents the data points. At the optimum the point sets become aligned and the correspondence is obtained using the maximum of the GMM posterior probability for a given data point ([187]). The algorithm minimizes the following energy function

$$E(f, \sigma^2) \quad = \quad \frac{1}{2\sigma^2} \sum_{m,n=1}^{M,N} P^{old}(m|x_n) \|x_n - (y_m + f(y_m))\|^2 \tag{6.23}$$

$$+ \quad T \log \sigma^2 + \frac{\lambda}{2} \|Lf\|^2, \tag{6.24}$$

where

$$P^{old}(m|x_n) = \frac{\exp\left(-\frac{1}{2}\|\frac{x_n - f(y_m)}{\sigma^{old}}\|^2\right)}{\sum_{k=1}^{M} \exp\left(-\frac{1}{2}\|\frac{x_n - f(y_k)}{\sigma^{old}}\|^2\right)} \tag{6.25}$$

and T is a constant. Different to the RPM algorithm, the authors in [187] use as regularizer $\|Lf\|^2$ the norm in the Hilbert space $\mathbb{H}^m$

$$\|f\|^2_{\mathbb{H}^m} := \int_{\mathbb{R}} \sum_{k=0}^{m} \|\frac{\partial^k f}{\partial x^k}\|^2. \tag{6.26}$$

Using calculus of variation it can be shown that given P^{old} the transformed points of shape Y have the form $f(Y) = Y + GW$, where $G \in \mathbb{G}^{M\times M}$ is a matrix of the form

$$g_{ij} = \exp\left(-\frac{1}{2}\|\frac{y_i - y_j}{\beta}\|^2\right) \tag{6.27}$$

and the matrix $W \in \mathbb{R}^{M\times 2}$ is the solution of the system

$$\left(G + \lambda\sigma^2 diag(P^{old}\mathbf{1})^{-1}\right) W = diag(P^{old}\mathbf{1})^{-1} P^{old} X - Y \tag{6.28}$$

and the optimal σ^2 is given by

$$\sigma^2 = \frac{1}{2N_p}\sum_{n=1}^{N}\sum_{m=1}^{M}\|x_n - f(y_m)\|^2 \tag{6.29}$$

$$N_p = \mathbf{1}^T P^{old}\mathbf{1} \tag{6.30}$$

The similarity to the ICP and RPM algorithm consists in that the CPD algorithm also alternates between the estimation of soft correspondences given by the matrix P^{old} and the estimation of a nonlinear global transformation of the form

$$f(Y) = Y + GW \tag{6.31}$$

which can be seen as the parametrization of regularized displacement vectors for all points in shape Y. The parameter σ^2 can be seen in the RPM algorithm, as the temperature parameter in an annealing procedure. However, instead of reducing the parameter by a deterministic schedule, the CPD algorithm calculates at each step the exact optimal value for it. A limitation of the CPD compared to the RPM algorithm is that the underlying nonlinear transformation model is only defined for the given point set and it is not clear how it can be continuously extended to the rest of the image.

The above mentioned probabilistic formulation using Gaussian Mixture Models (GMM) for shape registration is not new and has also been used in prior works. For instance, in [43] it was shown that for the RPM algorithm the alternation between correspondence and transformation estimation is equivalent to the Expectation Maximization (EM) algorithm for GMM, where one shape is treated as GMM centroids with equal isotropic covariances and the other shape is treated as data points. Whereas methods for rigid transformation like [124, 258, 55, 159, 175, 160] also formulate point set registration as a maximum likelihood estimation problem to fit the GMM centroids to the data points, earlier works on non-rigid point set registration [113, 208] also used the probabilistic formulation, where the GMM centroids were uniformly positioned along the contours (using splines to model them).

Finding Correspondences as the Solution of an Assignment Problem

The problem of finding correspondences between two point-set represented shapes has also been solved using an optimal assignment formulation. Assignment problems deal with the question of how to optimally assign n items (in the present case points of a shape) to n other items (the points in the second shape) [33].

For instance, the authors in [13] formulated the problem of matching two shapes as a linear assignment problem (LAP)

$$\min_x \sum_i^n \sum_{j=1}^n c_{ij} x_{ij} \tag{6.32}$$

$$s.t. \qquad \sum_{j=1}^n x_{ij} = 1,\ (\forall j = 1, \cdots, n) \tag{6.33}$$

$$\sum_{i=1}^n x_{ij},\ (\forall i = 1, \cdots, n) \tag{6.34}$$

$$x_{ij} \in \{0, 1\} \tag{6.35}$$

where the term c_{ij} refers to the cost of matching the shape descriptors of the points x_i and y_i. The corresponding constraints enforce the binary assignment matrix x to be a permutation (one-to-one matching). In this formulation no relation or structure in the shape (besides the information contained in the descriptors) is being taken into account for finding the correspondences resulting in a linear objective function.

However, there is also a class of algorithms which rely on quadratic objective functions. These methods rely on the solution or approximation of a *quadratic assignment problem* (QAP) (or modifications of it). In its general form, a QAP can be modeled as a quadratic integer program of the form:

$$\min_x \sum_{i,j=1}^n \sum_{k,l=1}^n \underbrace{a_{ik} b_{jl}}_{:=d_{ik;jl}} x_{ij} x_{kl} + \sum_{i,j} c_{ij} x_{ij} \tag{6.36}$$

$$s.t. \qquad \sum_{j=1}^n x_{ij} = 1,\ (\forall i = 1, \cdots, n) \tag{6.37}$$

$$\sum_{i=1}^n x_{ij} = 1 \tag{6.38}$$

$$x_{ij} \in \{0, 1\}\ (\forall i, j = 1, \cdots, n), \tag{6.39}$$

where the matrix d refers to the matching cost between two pairs of points and the constraints in (6.37) enforce one-to-one correspondences between the points in both shapes (however, this constraint can also be relaxed to enforce many-to-one or many-to-many matching). The problem (6.36) can also be equivalently formulated as

$$\min_z z^T D z + c^t z \tag{6.40}$$

$$s.t. \qquad Az = b, z \in \{0, 1\}, \tag{6.41}$$

where now the indicator variable z is such that $z_{ia} = 1$, if point y_i from one image is matched to the point x_a from the other image or otherwise zero. Although these methods are more descriptive than a LAP problem due to the quadratic cost which uses the structure

of the shape, they are NP hard to solve. Therefore, efficient algorithms must look for approximating the solution (e.g. [17, 148, 52, 101, 217, 271, 271, 71, 216]).
For instance, [17] defines a cost matrix $d_{ik;jl}$ which penalizes the change of direction and length between the pair of points y_i, y_j during matching. The problem (6.36) is then approximated by specifying a linear bounding problem of the quadratic term and thereafter a local gradient descent is used to find a locally minimal assignment.
A simple, yet very effective and widely used approach is the spectra matching algorithm of [148] which solves the relaxed variant

$$\max_x x^T Mx, \ s.t. \ x^t x = 1 \tag{6.42}$$

by calculating the first eigenvectors of the matrix M. Then by heuristically selecting the matches with highest eigenvalues, a binary matching is obtained. This algorithm was improved in [149], resulting in the *Integer Projected Fixed Point Algorithm* (IPFP). The intuition behind the algorithm is that at every iteration the quadratic term $x^T Mx$ is approximated by a first-order Taylor expansion around the current solution. This approximation is then maximized within the discrete domain of problem (6.42) [150].

6.3 Limitations of Current Approaches

From the analysis of previous methods, several conclusions can be sketched

Necessity of Local Morphological Analysis

Procrustes analysis [70] is an example of a class of methods which is able to infer a global transformation model to describe the deformation between two shapes and at the same time is capable to infer a global similarity measure between them. Furthermore, the idea of using a *single* global transformation for describing the deformation is also shared by methods like the ICP [40, 19] which uses a single affine transformation. Moreover, the RPM [44] or CPD [187] algorithms also use a single global non-linear transformation to transform the entire shape. However, using a single global transformation is not possible to give insights into how local structures in the shape transform. This still remains true if global measures like the bending energy of a TPS as described in Eq. (6.10) is considered, since it only yields a global numerical value about the distortion energy and thus lacks any local information.
In contrast, mesh-grid based methods like Thompson's Euclidean grids [231] or more recently finite element methods (e.g [25, 186, 132, 152, 42]) do estimate local transformations for each of the blocks in the grid. However, these methods are not able to reason whether a group of blocks can be described by the same transformation parameters. For instance, if an articulated object is considered, mesh-grid methods will break each of the articulations into several grids and use a single transformation to register each of them. Thus, this class of methods will ignore the fact that an articulation can probably be registered using a single affine transformation. On the contrary, piecewise affine models yield a good framework for achieving both, describing the global transformation of the shape change and at the same time describing the inner structure of the object. However, many piecewise affine models for registration still require manually selecting the different structures that are affine-transformed. Thus, automatically inferring the object structure is still an

open question. Further methods which use piecewise affine models are [253, 28]. However, this is done within the context of discrete motion segmentation which considers a different problem to the one of shape analysis.
From the above, the challenge remains to develop a method which is capable of describing the global transformation of the shape which at the same time can give insights into the true structure (and not a grid-quantization) of the underlying object.

Automatic Complexity Adaptation

Non-linear deformations models like Thin-Plate-Splines or the point-wise displacement parametrization of the CPD algorithm are powerful methods for describing complex transformations. Normally, the complexity of these models is regulated by a global parameter. For instance, if the parameter λ in Eq. (6.17) for the TPS is set too high, a rigid transformation is obtained. On the contrary, if it is set too low, the global transformation will overfit due to noise. Therefore, a model which automatically adapts its complexity according to the current deformation is required. Whereas a rigid object should be described with a single linear transformation, the model should adapt the complexity according to the deformation. In other words the challenge is to follow the Occam's razor (or in latin *lex parsimoniae*), a principle which states that one should use simpler models and increase the complexity if a greater explanatory power is required.

Joint Model

The different piecewise methods which were introduced in this chapter, can be divided as (a) models that focus on solely calculating the transformation model, (b) models that simultaneously estimate the correspondences and the deformation model and (c) models that first estimate the correspondences and only afterwards is the transformation model (e.g. [17, 150]) estimated. Therefore, it remains open how to jointly obtain the correspondences, estimate the complexity of the transformations, discover the local structure of the objects and estimate the overall transformation of the shapes using a single optimization procedure. This challenge will be discussed in the next chapter.

CHAPTER 7

BEYOND GLOBAL TRANSFORMATIONS - MORPHOLOGICAL ANALYSIS FOR INVESTIGATING ARTISTIC REPRODUCTIONS

This chapter introduces a methodology for solving the problem introduced in section 1.5.2: Is it possible to develop a method, and thus an algorithm, capable of automatically finding the appropriate deformation between two similar objects and infer at the same time the structure and the complexity of such a deformation between objects? Understanding what an object is means inter alia to understand all the properties that characterize it. Shape is one of the most important features that characterizes an object (s. Sec. 1.4). It may, however, change between object instances though remaining distinguishable by the human mind as being a *common* shape to both objects despite the changes. What remains the same and what changes? The present chapter aims at developing a computer vision system able to answer this question. We focus on the analysis of artistic objects, since the analysis of object shapes within art history is also important to reveal the stylistic implications of artworks and thus, such analysis may help to disclose the historical influences in the creation and reproduction of art. This analysis shows once more the fruitfulness of the interaction between computer vision and cultural heritage. The present results were submitted for publication in [185].

7.1 Introduction

Although some stylistic movements in art like impressionism or pointillism define themselves by color, shape has been the predominant way to perceive an artwork. The theory about primacy of shape can be traced back to Giorgio Vasari (1511-1574) who propagated the line drawing as the predominant technique of all visual arts. His use of the term *disegno* (conceptual design) can be read as the assignment of ideas to shapes. This "shaped idea" is represented through shapes in preparatory drawings, in the artwork itself as well as in drawn reproductions. Based on this observation, changes in shape between artworks and their reproductions, or preparatory drawings can be associated with changes in ideas and concepts which reveal artistic choices and stylistic variations. Thus, the analysis of these changes helps art historians to understand the impact of discourses and historical influences in the creation and reproduction of art. However, in many cases these alterations between shapes are very subtle and thus it becomes extremely difficult, even for trained eyes, to determine the nature and extent of the deformations suffered by different parts within an artwork. The automatic solution of such shape analysis poses an ambitious computer vision task and its solution is the focus of the present chapter. The nature of the artwork deformations that are analyzed in this work arise either due to deliberate alterations or due to geometrical errors accumulated during the drawing process. For instance, a typical example for a deliberate alteration between a preparatory drawing and the finished work is a conceptual change that induces alterations in the relative position of extremities in a human pose. Thus, in this case it is of interest for art historians to recognize the parts that feature the same transformation and determine to which extent these parts differ from other regions in the image. The second class of deformations is more subtle and is related to the drawing process itself. Copying in many cases was done by placing a thin, tracing paper on top of the original, and sketching the contours. Movements of the semi-opaque sheet by the artist induced slight modifications in the reproduction. Whereas parts that were reproduced at the same time shared the same transformation, sheet movements induced a different transformation for the rest of the reproduction. Therefore, the system presented in this chapter addresses the description of such overall nonlinear deformations at the same time that give insights about the structure of different local deformations present in the image. In the following, a more detailed description of the characteristics of our model is given:

Piecewise Transformation Model

Shape transformation models within computer vision can be classified into linear and nonlinear models. Since *global* linear models cannot be used for describing complex shape changes due to their limited description power, a common choice for describing nonlinear changes has been the usage of splines like the TPS [44]. In this case, a TPS consists of both a global affine transformation part and a non-linear parametrization based on radial basis functions, where the number of parameters required for its estimation is in the order of the number of points in the shape (s. Sect. 6). Moreover, the complexity of such a model is regulated by a single parameter for the entire shape which is required to be manually set. If this parameter is set too low, the registration becomes instable since noise in the shape or in the correspondence assignments between points yields an over-fitted transformation. In contrast, a high regularization parameter induces rigid transformations incapable of describing the changes for the entire shape. Furthermore, the global nature of this parameter makes it impossible for the model to locally adapt its complexity according to the shape deformation. Therefore, the present chapter presents a piecewise linear registration model that adapts the complexity of each component according to the shape deformation in the underlying region. Moreover, the assignment of regions in the shape to different model components induces a clustering which is used in turn to visualize the structure and geometry of the deformation introduced by the artist during the reproduction procedure.

Automatic Complexity Estimation

However, a challenge of using piecewise linear models consists also in automatically determining the complexity of the model, that is, the number of affine components required for registration. In the absence of prior knowledge about the shape, this question represents a particularly important part of the analysis. Nonetheless, an indispensable requirement for selecting the number of components is the robustness of the registration solution. The present paper considers this robustness or stability from a statistical point of view, that is a stable registration solution for a given number of components is a solution that is reproducible on different subsampled versions of the shape and not too sensitively dependant on the sample set at hand. For instance, inferring too many affine components (or clusters) will lead to very similar affine transformations and points will arbitrarily be assigned to them due to sample fluctuations. However, if too few clusters are selected structures in the shape that should be kept separate will be mixed. Therefore, the "correct" number of transformations is defined as the number which yields the most stable solution capable of handling the trade-off between too rigid transformations and an overparametrization of the transformation model.

Automatic Assignment of Affine Components

The second challenge of using piecewise affine models consists of determining not only the correspondence between shapes but also which parts in the shape can be assigned to the different affine components. Whereas piecewise affine models like [199, 48, 4, 188, 116] simplify the task by manually setting the spatial domain of each of the affine components, the methods in [181, 180, 253] assume to have the correspondence between shapes. In this chapter we present a model that simultaneously solves three tasks: (i) infer the point-correspondences between both shapes, (ii) identify the groups in the image which share the same transformation, and (iii) estimate the transformation of these groups. Tasks (ii) and (iii) are intrinsically related. The reason is that whereas a group is defined as the set of points in a shape that can be described by a certain affine component, this affine component is estimated using the points that define the group. An additional difficulty also becomes evident noticing that whereas a group is a *discrete* set of points, the corresponding affine parameters are *continuous* resulting in a complex mixed-integer problem.

Historical Analysis of Image Reproductions

The last part of the present chapter analyzes prominent reproductions from different periods of art history. At first, images coming from the Codex Manesse illustrated between c. 1305 and c. 1340 in Zürich and their reproductions commissioned by Bodmer/Breitinger in 1746/1747 are considered. This image collection is important in art history since the Codex Manesse is the single most comprehensive source of Middle High German Minnesang poetry [36] and represents an outstanding source for understanding the visual interpretation of the Middle Ages in early modern and modern times. Whereas the tracings from book illustrations like the reproductions of the Codex Manesse exhibit only slight changes, the differences between a drawing and a mural painting are obviously greater. Therefore, we also analyze parts of Michelangelo's ceiling fresco in the Sistine Chapel (1508-1512) with sketches, which were made in the artist's surroundings, probably after Michelangelo's own preparatory drawings or by Dutch artists after the original artwork had been completed.

7.2 Novelty of the Approach

In [181, 180] we introduced for the first time a method for analyzing the reproduction process of artworks. However, both methods featured different limitations which we briefly describe in the present section. In section 7.3 we will then introduce a new approach with the intention of overcoming the prior limitations of [180, 181].

Greedy Clustering

The problem analyzed in [181] can be considered as a simplified version of the full task introduced in section 7. The reason is that the authors *manually* select both points along the contours and the correspondences between the point sets. Such information is usually not available for large-scale tasks or is tedious and expensive to obtain. In the following, shapes are represented through landmark points. The shape of the original artwork is referred to by the set $\{x_i\}_{i=1}^m$ and $\{y_i\}_{i=1}^n$ for the reproduced shape. Furthermore, $T(\vartheta_i)$ denote an affine transformation that is estimated in an iterative fashion for each point y_i and a neighborhood of it: if $T_i^1 := T(\vartheta_i^1)$ is the initial affine transformation estimated for point y_i using a non-collinear set of 12 neighbors, the refined transformation in the next iteration is given by

$$\begin{aligned} \vartheta_i^{k+1} &:= \vartheta_i^k \cup \vartheta_{\operatorname{argmax}_j E_{ij}^k} && (7.1) \\ T_i^{k+1} &= T(\vartheta_i^{k+1}) && (7.2) \\ \mathcal{X}_A^{k+1} &:= \mathcal{X}_A^k \setminus \vartheta_i^{k+1}, && (7.3) \end{aligned}$$

where

$$E_{ij}^k := \left| \left\{ s \mid \left\| T(\vartheta_i^k \cup \vartheta_j^1) y_s - x_s \right\|_2 \le \epsilon \right\} \right| \tag{7.4}$$

and

$$C_i^k := \left\{ j \mid \frac{1}{|\vartheta_j^k|} \sum_{s: y_s \in \vartheta_j^k} \left\| T_i^k y_s - x_s \right\|_2 \le \epsilon \right\}. \tag{7.5}$$

Thus, the underlying idea consists of updating the neighborhood set and the transformation iteratively (eq. 7.1) by searching for points in the shape that can be explained by the current transformation (eq. 7.5 and eq. 7.4). Once an affine transformation is calculated for every point in the shape, a further clustering is applied to obtain a reduced number of transformations capable of registering both shapes. Therefore, the similarity measure is used

$$\Delta_{ij} = \beta_1^{-1} d_T(y_i, y_j) + \lambda \beta_2^{-1} d_C(y_i, y_j), \tag{7.6}$$

where $d_C(y_i, y_j)$ is an Euclidean contour distance and

$$d_T(y_i, y_j) := \frac{1}{2}\left(\|T_j y_i - T_i y_i\| + \|T_j y_j - T_i y_j\|\right) \tag{7.7}$$

Several limitations of this approach become apparent:

- The approach lacks a unified framework: Two different clustering procedures are independently used. One is applied for estimating affine transformations (eq. 7.1) and the other reduces the number of transformations (using eq. 7.6) required for registration in order to avoid an overfitting.
- The correspondence problem is not approached. Furthermore, a method to automatically extract edges, locate, and match landmark points is missing.
- The complexity of the piecewise affine model is manually set.
- The final clustering uses an Euclidean-distance term (eq. (7.6)) to force the compactness of the groups. This term introduces a bias in the result since the objective is to find groups of transformations that register the artwork and its reproduction.

A Deterministic Annealing Approach

The first two limitations enumerated in the last subsection were approached in [180], where a single optimization problem was formulated in order to estimate the affine transformations and the resulting groups. There the shape contours, the landmark points, and the correspondences were automatically inferred. However, the estimation of the point correspondences was carried out independently of the other tasks.

Formally, the approach in [180] consisted in the estimation of a binary data assignment matrix $M \in \mathbb{B}^{n\times k}$ of n points to k groups at the same time that different affine transformations $T^\nu \in \mathbb{R}^{3\times 3}$ $(\nu = 1, \ldots, k)$ for each group were calculated. In this case, the matrix M is defined such that $m_{i\nu} = 1$ only if point $\mathbf{x}_i$ is assigned to group ν.

To find both the data partition matrix M and the affine transformations T^ν, local affine transformations T_i were first calculated. These transformations are different from T^ν. While the former were calculated using only a small neighborhood around each landmark point (12 non-collinear points) and were kept fixed, the latter transformations of groups T^ν corresponded to the deformations present in the reproduction process and were optimized together with M. In a next step, the energy function $E(M, T^\nu)$ was formulated

$$\min_{M,T^\nu} E(M, T^\nu) = \min_{M,T^\nu} \frac{1}{2}\sum_{\nu=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{M_{i\nu}M_{j\nu}}{p_\nu}a_{ij} + \sum_{\nu=1}^{k}\sum_{i=1}^{n} M_{i\nu}r_{i\nu} \tag{7.8}$$

$$s.t. \quad \sum_{\nu=1}^{k} M_{i\nu} = 1 \ (\forall i = 1, \cdots, n), \ \ M_{i\nu} \in \{0, 1\} \tag{7.9}$$

$$a_{ij} := \frac{1}{Z}\left(\|T_j x_i - T_i x_i\| + \|T_j x_j - T_i x_j\|\right) \tag{7.10}$$

$$r_{i\nu} := \frac{1}{Z_i}\left(\lambda_2 ||T^\nu x_i - y_i||_2^2 + (1-\lambda_2)||T_i x_i - T^\nu x_i||_2^2\right), \tag{7.11}$$

which was solved in turn using a coordinate descent approach based on deterministic annealing. For this, the idea was to relax the matrix M to be a continuous valued matrix

$\hat{M}$ in the interval of [0 1] and introduce a $M \log M$ entropy barrier function, which allowed fuzzy partial assignments of data points to groups in the matrix $\hat{M}$. This term was controlled in turn by a temperature parameter β. For $\beta \to 0$ a discrete relaxed energy function $\hat{E}(\hat{M}, T^\nu; \beta)$ was defined as follows

$$\min_{\hat{M}, T^\nu} \hat{E}(\hat{M}, T^\nu; \beta) \quad := \quad \frac{1}{2} \sum_{\nu=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\hat{M}_{i\nu} \hat{M}_{j\nu}}{p_\nu} a_{ij} + \sum_{\nu=1}^{k} \sum_{i=1}^{n} \hat{M}_{i\nu} r_{i\nu} \tag{7.12}$$

$$+\beta \sum_{\nu=1}^{k} \sum_{i=1}^{n} \hat{M}_{i\nu} \left(\log \hat{M}_{i\nu} - 1 \right) \tag{7.13}$$

$$s.t. \qquad \sum_{\nu=1}^{k} \hat{M}_{i\nu} = 1 \; (\forall i = 1, \cdots, n), \quad \hat{M}_{i\nu} \in \{0, 1\} \tag{7.14}$$

As described in [269], the minima of $E(M, T^\nu)$ and $E(\hat{M}, T^\nu; \beta)$ all coincide in the limit $\beta \to 0$ if the matrix (a_{ij}) is negative definite. This was obtained by adding a sufficiently large term to its diagonal without altering the structure of the minima of $E(M, T^\nu)$. The linear constraints were imposed by adding a Lagrange multiplier term obtaining the Lagrange function

$$L(\hat{M}, \mu) \quad := \quad \frac{1}{2} \sum_{\nu=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\hat{M}_{i\nu} \hat{M}_{j\nu}}{p_\nu} a_{ij} + \sum_{\nu=1}^{k} \sum_{i=1}^{n} \hat{M}_{i\nu} r_{i\nu} \tag{7.15}$$

$$+\beta \sum_{\nu=1}^{k} \sum_{i=1}^{n} \hat{M}_{i\nu} \left(\log \hat{M}_{i\nu} - 1 \right) + \sum_{i=1}^{n} \mu_i \left(\sum_{\mu=1}^{k} \hat{M}_{i\nu} - 1 \right) \tag{7.16}$$

The Lagrangian function is a sum of a convex function $E_{vex}(\hat{M}) = \beta \sum_{\nu i} \hat{M}_{i\nu} \log \hat{M}_{i\nu}$ and a concave part $E_{cave}(\hat{M}) = (1/2) \sum_{\nu ij} \frac{\hat{M}_{i\nu} \hat{M}_{j\nu}}{p_\nu} a_{ij} + \sum_{\nu i} \hat{M}_{i\nu} r_{i\nu}$. Using this fact, the CCCP algorithm ([270]) was used. This procedure effectively minimized the energy function using the following update rule

$$\beta \left(1 + \log \hat{M}_{i\nu}^{t+1} \right) = -\frac{1}{2} \sum_j \hat{M}_{j\nu}^t \frac{a_{ij}}{p_\nu} - r_{i\nu}, \tag{7.17}$$

after setting to zero the derivative of (7.15) with respect to μ_i. By substituting it into equation (7.17) and solving for $\hat{M}_{i\nu}^{t+1}$ the following update rule was obtained

$$\hat{M}_{i\nu}^{t+1} = \frac{\exp\left(\beta \left(-\frac{1}{2} \sum_j \hat{M}_{j\nu}^t \frac{a_{ij}}{p_\nu} - r_{i\nu} - 1 \right) \right)}{\sum_\nu \exp\left(\beta \left(-\frac{1}{2} \sum_j \hat{M}_{j\nu}^t \frac{a_{ij}}{p_\nu} - r_{i\nu} \right) \right)} \tag{7.18}$$

After each update step (7.18), the affine transformations were recalculated using the Levenberg-Marquardt algorithm:

$$T^\nu = \arg\min_{T^*} = \sum_{i}^{N} \hat{M}_{i\nu}^{t+1} \left(\lambda_2 ||T^* x_i - y_i||^2 + (1 - \lambda_2) ||T_i x_i - T^* x_i||_2^2 \right) \tag{7.19}$$

In order to initialize the algorithm, the initial matrix $\hat{M}^0$ was obtained by running a fuzzy c-means algorithm using the Euclidean distance between points x_i ([20]) and $\hat{M}^0$ was taken to be the resulting fuzzy assignment matrix.
Although this method is capable of jointly solving for the groups and the transformations the usage of DA for its solutions is problematic. The problem arises at the beginning of the optimization procedure when the temperature parameter is high, since all points are assigned to every initial affine transformation with almost the same probability. Thus, all parameters become equal when the transformations are updated. A second limitation (also shared by [181]) is that the energy function to be minimized includes a Euclidean-distance term, which forces the compactness of the groups and introduces a bias as described above. Furthermore, [180] assumes the initial correspondences as fixed and thererfore are not actualized during the optimization procedure. The method described in the next section replaces the deterministic annealing by introducing a linear program (LP) formulation. Moreover, the Euclidean-distance term in the energy function is also eliminated. In addition to this, our method also updates the correspondences between shapes along with the groups and the transformations within the same optimization procedure.

Related Fields

Within the field of sparse motion segmentation, the authors in [253] presented a procedure for decomposing videos into similarly moving layers. The scene was firstly divided into a regular grid and an affine transformation was calculated for each block. The method estimated affine motion models for segments on a regular grid. However, due to clutter and missing contours, accurate estimation of small and continuous deviations in transformations cannot be estimated with this approach. In [63], a regularized energy function was minimized with Graph-Cuts ([30]) which also included a pairwise regularization and thus a bias in the result. Thus, this regularization leads in practice to a poorer registration quality since parts in the shape belonging to different model components are mixed. Furthermore, the authors of [133] presented a LP formulation of a central clustering in which the number of clusters is determined indirectly by a hard to determine penalty term for each data point. Lazic et al. [141] also indirectly determined the number of clusters through the weighting of the different random subsampled linear subspaces. Normally, (rigid) motion segmentation can be seen as an application of the more general task of subspace segmentation [141, 264]. This latter task commonly assumes that the data points lie on several distinct *linear subspaces* [114, 264, 58, 246, 126]. However, the linearity assumption does not hold in our setting: Whereas shape points lie in a 2D vector space, each of the shape parts that were similarly altered by the artist are represented through elements of the affine group. Therefore, the task consists not only of clustering points which define a linear subspace but three tasks needing to be jointly solved: the correspondence between both shapes, the groups in the image which share the same transformation, and the estimation of the transformations of those groups.
In the field of computer graphics Sýkora et al. [230] embedded each shape in a lattice consisting of several connected squares and registered them by estimating a rigid transformation for every square. Since the registration is only on the level of rigid squares, a grouping into flexibly shaped regions with related modifications is not part of this contribution. Furthermore, the authors of [230] are not able to handle deformations which do not preserve local rigidity (e.g s scaling or shear) and it requires a significant overlap between shapes for registration. Additionally, in our setting background clutter creates distractors that need to be handled, whereas the method of [230] is only applied to cartoons without

any clutter. Another interesting related work is [48], which presented a piecewise affine regularization method for medical image registration. The drawback of this method is that the affine- registered areas required to be estimated manually by the user. Related to piecewise affine registration, the authors of [116] recently introduced a matching algorithm based on affine transformations calculated on a triangulation of the shape. In this case, to match articulated objects it is required to manually select the groups and their articulation in order to match the scene images. Two different works which are related to estimating transformations between artworks are [39, 239]. While [39] tries to ensure consistent perspective in art images, [239] aims to dewarp image reflections shown in convex mirrors within very specific paintings. Common non-linear registration algorithms like [44] or [187] are also not suited to the purpose of the present task. Whereas [44] uses a Thin Plate Spline (TPS) to model the transformation, [187] estimates a displacement vector for each point in the shape. In both cases these models introduce artifacts in the registration as observed in [180], which is undesirable for art comparison.

7.3 Automatic Estimation of Transformations, Groups and Correspondences

In the present chapter shapes are represented through landmark points (given in homogeneous coordinates) which are regularly sampled along extracted contours of the corresponding image in an automatic manner (s. Sec. 7.4.4 for more details). Thus, the shape of the original artwork is referred to with the matrix $X \in \mathbb{R}^{m\times 3}$ and with $Y \in \mathbb{R}^{n\times 3}$ the reproduced shape.

7.3.1 Problem Statement

The main challenge consists of simultaneously solving three tasks. Firstly, the correspondences between both shapes have to be inferred. Secondly, the groups in the image which share the same transformation need to be found and finally, the transformations of those groups and thus the overall deformation model needs to be estimated. The missing groups correspond to image regions which are reproduced similarly by the artist. Therefore, each of these groups is modeled through an affine transformation capable of transforming the group from the reproduction into the original painting. The advantage of using a piecewise-affine transformation model is that it allows to describe a non-linear transformation in a more parsimonious manner, that is, less parameters are required for describing the overall transformation. At the same time, the components in the model associated with different regions in the shape give insights about the structure and geometry of the artistic deformation.

Formally, the problem consists of estimating a binary data assignment matrix $C \in \mathbb{B}^{n\times m}$ of n points belonging to the first shape to m points in the second shape. At the same time, a binary matrix $M \in \mathbb{B}^{n\times k}$ of n points to k groups needs to be calculated together with different affine transformations $T^{\nu} \in \mathbb{R}^{3\times 3}$ $(\nu = 1, \ldots, k)$ for each group. Thus, the overall registration error made by a solution $(M, C, T^1, \cdots, T^k)$ can be written as:

$$E_{reg} := \sum_{i,\nu=1}^{n,k} M_{\nu i} \left(\underbrace{\sum_{j=1}^{m} C_{ij} \|x_j - T^{\nu} y_i\|^2}_{=:r_{\nu i}} \right) . \tag{7.20}$$

An important observation is that although the global deformation between both artworks is expected to be non-linear, regions between both images that were copied without any or little alteration by the artist are transformed homogeneously and therefore, these parts can be described using a single affine transformation. Thus, for any two points y_i, y_j within such an affine-transformed shape part together with their respective correspondent points x_a, x_b, the distortion between the vector from y_i to y_j and the vector from x_a to x_b is expected to be small (and minimal in the presence of a rigid transformation). Similar to [17], this distortion can be measured by

$$\begin{aligned} d(y_i, y_j; x_a, x_b) &:= \gamma d_a(y_i, y_j; x_a, x_b) \\ &\quad + (1-\gamma) d_l(y_i, y_j; x_a, x_b), \end{aligned} \tag{7.21}$$

$$d_a(y_i, y_j; x_a, x_b) := \left(\frac{\alpha_d}{|s_{ij}|} + \beta_d \left| \arcsin\left(\frac{\hat{s}_{ab} \times s_{ij}}{|\hat{s}_{ab}||s_{ij}|} \right) \right| \right), \tag{7.22}$$

$$d_l(y_i, y_j; x_a, x_b) := \frac{||s_{ij}||\hat{s}_{ab}||}{(|s_{ijf}| + \sigma_d)}; \tag{7.23}$$

$$s_{ij} := y_i - y_j, \ \hat{s}_{ab} := x_a - x_b. \tag{7.24}$$

Whereas the first term $d_a(y_i, y_j; x_a, x_b)$ penalizes the change in direction, the second term $d_l(y_i, y_j; x_a, x_b)$ penalizes the change of length between two pairs of points in both shapes. The constants $\alpha_d = \beta_d = \sigma_d = 0.5$ allow more flexibility for nearby points, and the constant $\gamma = 0.3$ weighs the angle distortion term against the length distortion term. We use this measure to further enforce the matching consistency between both shapes and thus the energy term (7.20) to be minimized is extended to:

$$\begin{aligned} \min_{M,T^\nu,C} E_{tot} := \sum_{i,\nu=1}^{n,k} M_{\nu i} \left(\sum_{j=1}^{m} C_{ij} \|x_j - T^\nu y_i\|^2 \right) + \\ \underbrace{\sum_{\nu=1}^{k} \sum_{i,j=1}^{n} \sum_{a,b=1}^{m} M_{\nu i} M_{\nu j} C_{ia} C_{jb} d(T^\nu y_i, T^\nu y_j; x_a, x_b)}_{=:E_{quad}} \end{aligned} \tag{7.25}$$

$$s.t. \sum_{\nu=1}^{k} M_{\nu i} = 1 \ (\forall i = 1, \cdots, n) \tag{7.26}$$

$$\sum_{i=1}^{n} C_{ij} = 1 \ (\forall j = 1, \cdots, n), \tag{7.27}$$

$$C_{ij} \in \{0,1\}, \ \ M_{\nu i} \in \{0,1\} \tag{7.28}$$

where k is the complexity of the piecewise model (i.e. the number of affine transformations desired for registration). This parameter will be set automatically based on the stability analysis described in section 7.3.3. Whereas the constraint (7.26) forces each point to be assigned to a single group, the constraint (7.27) ensures a many-to-one matching between both point-sets yielding robustness in cases of missing points. Important to remark is that whereas the authors of [17] minimized the pairwise distortions for all points in the shape together, our model minimizes the pairwise distortions within each of the groups defined through the matrix M.

7.3.2 Optimization Strategy

The general setting of jointly solving for M, C, T^{ν} is hard. This is reflected in the above problem formulation (7.25), where solving for the matrix C exactly is already NP-hard [17]. A practical solution to minimize the above energy is to assume an alternating procedure. Departing from an initial solution, the above energy function is reduced by first calculating the matrix M and the transformations T^{ν} (assuming the matrix C is given) and solving for the matrix C (assuming M, T^{ν} are given) thereafter. This procedure is iterated until the matrix C and M do not change.

Problem Formulation using a Superset of Affine Transformations

Estimating the matrix M and the different affine transformations T^{ν} (given the matrix C) are closely interrelated problems and their solution poses a challenging issue. Whereas T^{ν} ($\nu = 1, \cdots, k$) can only be estimated when the assignment of points to k groups (given by the matrix M) is known, each of the groups ν is defined by the fact that all points within it can be registered using a single affine transformation T^{ν}. Thus, the rationale of our previous work [180] was to approach this problem by first proposing a *single* initial clustering (i.e. a matrix M) based on the Euclidean proximity of the shape points. Thereafter, based on this matrix, the estimation of the affine transformations T^{ν} was alternated with the actualization of matrix M until local convergence was reached. However, this procedure turned out to be very susceptible to the initialization of the matrix M. We show this fact in Fig. 7.1 (e) where a textitsingle initial k-tuple of affine transformations led to a wrong clustering, where parts in the shape corresponding to different affine deformations were mixed into the same group. This paper studies an orthogonal approach for solving the aforementioned problem leading to better results as shown Fig. 7.1 (b) (s. experimental section for more details). Instead of proposing an initialization for the matrix M or a *single* k-tuple of affine transformations and thus risking a wrong initialization, we construct a large superset of affine transformations

$$T_{\text{pool}} := \{T^{\nu} \,|\, T^{\nu} \in \mathbb{R}^{3\times 3},\, \nu = 1, \cdots, l\}, \tag{7.29}$$

where $l >> k$. For this purpose the shape Y is subdivided into non-overlapping small segments, each of them containing at least 6 non-collinear points. For each segment an affine transformation is estimated and added to the superset T_{pool} (we assume to have an estimate of matrix C). Thereafter, each segment is merged with its nearest neighbor and an affine transformation is calculated for the merged segment, which in turn is added to T_{pool}. For the nearest neighbor estimation, the distance between two segments is defined as the Euclidean distance between their centers of mass (i.e. the average of the segment points). This merging is repeated until the whole shape is merged into a single segment. Thereafter, using this superset T_{pool} our algorithm optimally selects a subset of k transformations that best register the shape and use these active transformations to estimate the matrix M. Based on this matrix the active transformations are then updated in turn. Thus, the original problem (7.25) is transformed into its final form:

$$\min_{M,W,C,T^\nu} \underbrace{\sum_{\nu=1}^{l} w_\nu \left(\sum_{i=1}^{n} M_{\nu i} r_{\nu i} \right)}_{=:E_{lin}(W,M,C,T^\nu)} + E_{quad} \tag{7.30}$$

$$s.t. \sum_{\nu=1}^{l} w_\nu = k, \tag{7.31}$$

$$n * w_\nu - \sum_{i=1}^{n} M_{\nu i} \geq 0 \ (\forall \nu = 1, \cdots, l) \tag{7.32}$$

$$w_\nu \in \{0, 1\} \tag{7.33}$$

plus the constraints (7.26)-(7.28). Here the binary vector $w_\nu = 1$ indicates that the ν-th element of the set T_{pool} is being used and otherwise $w_\nu = 0$. Whereas the constraint (7.31) guarantees to obtain the desired number of transformations k, the constraint (7.32) avoids the assignment of points to inactive transformations $w_\nu = 0$. This becomes clearer by remarking that constraint (7.32) is fulfilled whenever the logical constraint $w_\nu = 0 \Rightarrow \sum_{i=1}^{n} M_{\nu i} = 0$ is met.

LP-based Solution for Transformations and Group Assignments

In this section we describe how to estimate the active transformations (i.e. the vector W), assign points to the corresponding transformations (through the matrix M) and update them afterwards (we assume to have the matrix C). This is a hard task due to the quadratic non-linear term E_{quad} in Eq. (7.30). Therefore, in praxis we focus only on the minimization of the term E_{lin}. Doing this is meaningful since this term controls the overall registration error defined in Eq. (7.20) which in the praxis we intend to minimize. A further difficulty is given by the binary constraints on M and W. For instance, if the elements w_ν are relaxed to $w_\nu \in$ [0, 1], the constraint $\sum_{\nu=1}^{n} w_\nu = k$ becomes a soft-constraint. Therefore, despite fulfilling this constraint more than k elements, w_ν can become greater than zero due to the relaxation. Thus, the constraint (7.32) will assign points to more than k transformations yielding a wrong solution to the joint problem. However, this last problem is alleviated if we adopt an alternate procedure to minimize E_{lin}. Firstly, the relaxed LP subproblems is solved:

$$\min_{W} \quad \sum_{\nu=1}^{l} w_\nu \left(\sum_{i=1}^{n} M_{\nu i} r_{\nu i} \right) \tag{7.34}$$

$$s.t. \quad \sum_{\nu=1}^{l} w_\nu = k \ (\forall i = 1, \cdots, n), \ \ w_\nu \in [0\,1]. \tag{7.35}$$

During the first iteration, all elements of matrix M are set to one and the transformations to build r are taken from T_{pool}. Since the solution W may contain more than k nonzero elements (due to the relaxation), we observe excellent results if only the k biggest elements are set to $w_\nu = 1$ and the rest to zero. Once the active transformations W are obtained, the second step consists in assigning points to these transformations by solving the LP:

$$\min_{M} \quad \sum_{i,\nu=1}^{n,k} M_{\nu i} r_{\nu i} \tag{7.36}$$

$$s.t. \quad \sum_{\nu=1}^{k} M_{\nu i} = 1 \ (\forall i = 1, \cdots, n) \tag{7.37}$$

$$M_{\nu i} \in [0\,1], \tag{7.38}$$

Here, the matrix $M \in \mathbb{R}^{k \times n}$ only indicates the assignment of points to the k active transformations. A further benefit of solving this subproblems is that the solution of (7.36) is always an integer solution, due to the constraint nature of M. Finally, after solving for the matrix M, new point assignments for the different groups are made. Thus, the active transformations can be actualized exactly by noting that given M and W, solving for the affine parameters result in a weighted least squares problem for each group:

$$\min_{T^\nu} \sum_{i=1}^{n} \sum_{j=1}^{m} \underbrace{(C_{ij}(M_{\nu i}))}_{:=(p^\nu)_{ij}} \|x_j - T^\nu y_i\|, \ (\forall \nu : w_\nu = 1) \tag{7.39}$$

which can be solved exactly ([206]):

$$N_P \ := \ \mathbb{1}1^T P \mathbb{1}1, \ \mu_x := \frac{1}{N_P} X^T P^T \mathbb{1}1, \ \mu_y := \frac{1}{N_P} Y^T P \mathbb{1}1 \tag{7.40}$$

$$\hat{X} \ = \ X - \mathbb{1}1 \mu_x^T, \ \hat{Y} := Y - \mathbb{1}1 \mu_y^T, \tag{7.41}$$

$$B \ := \ \left(\hat{X} P^T \hat{Y}\right) \left(\hat{Y}^T \mathrm{diag}(P \mathbb{1}1) \hat{Y}\right)^{-1} \tag{7.42}$$

$$t \ := \ \mu_x - B \mu_y \tag{7.43}$$

$$T^\nu \ := \ \begin{pmatrix} B & t \\ 0 & 1 \end{pmatrix} \tag{7.44}$$

For clearness in the notation, we have dropped the index ν out of the matrix P^ν defined in Eq. (7.39). Furthermore, diag(v) refers to the diagonal matrix built up using the elements of the vector v, $\mathbb{1}1$ being a vector containing 1 in all entries. Finally, this exact estimation of the affine transformations is an improvement over [181, 180], where the transformations are only approximated using Levenberg-Marquardt.

Finding Correspondences

We first describe how to estimate the correspondence matrix C between shapes Y and X assuming the knowledge of the groups M and the transformations T^ν (i.e. transformations

T^ν where $w_\nu = 1$) Thus, the problem (7.30) can be formulated as

$$\min_C \sum_{\nu=1}^{k} w_\nu \left(\sum_{i=1}^{n} M_{\nu i} r_{\nu i} \right) + \tag{7.45}$$

$$\sum_{\nu=1}^{l} \sum_{i,j=1}^{n} \sum_{a,b=1}^{m} M_{\nu i} C_{ia} M_{\nu j} C_{jb} d T^\nu y_i, T^\nu y_j ; x_a, x_b$$

$$\sum_{i=1}^{n} C_{ij} = 1 \ (\forall j = 1, \cdots, n), \ \ C_{ij} \in \{0,1\} \tag{7.46}$$

$$\tag{7.47}$$

This problem can be alternatively formulated as

$$\min_z \qquad \sum_{\nu=1}^{k} z^T D^\nu z \tag{7.48}$$

$$s.t. \qquad Az = 1, \ \nu = 1, \cdots, k \tag{7.49}$$

$$z \in \{0,1\}. \tag{7.50}$$

In this case, z is an indicator vector such that $z_{ia} = 1$ if point y_i from one image is matched to point x_a from the other image and zero otherwise. In this formulation, the information of matrix C is included in the vector z. Furthermore, each matrix D^ν contains the values $d(T^\nu y_i, T^\nu y_j; x_a, x_b)$ in all entries corresponding to group ν (i.e. $M_{\nu i} \ = 0$) and zero otherwise. Whereas the diagonal of D^ν consists of the linear terms of equation (7.45), the many-to-one constraints of matrix C are expressed by the matrix A.
Solving for each group independently consists of subdividing the above formulation (7.45) into smaller problems. This local formulation is given if first the vector u^ν is defined. This vector contains all entries of vector z corresponding to all points y_i belonging to group ν (i.e. all entries of the form $z_{i\bullet}$ for which $M_{\nu i} = 1$, $w_\nu = 1$). Using this vector we obtain the following local problems:

$$\min_{u_\nu} \qquad u^{\nu T} D^\nu_{|u^\nu} u^\nu \tag{7.51}$$

$$s.t. \qquad A_{|u^\nu} u^\nu = 1 \tag{7.52}$$

$$u^\nu \in \{0,1\}, \ (\forall \nu : w^\nu = 1) \tag{7.53}$$

where $D_{|u^\nu}$ is the submatrix of D^ν containing only pairwise distortions related to points belonging to group ν (the non-zero submatrix of D^ν in Eq. 7.48) and $A_{|u^\nu}$ is the many-to-one constraint submatrix of A for the corresponding points. Each of the subproblems in Eq. (7.51) is then approximated using the *Integer Projected Fixed Point* (IPFP) algorithm for graph matching ([149]) described in chapter 6. To estimate the initial matrix C required by the IPFP algorithm, both shapes X and Y are registered using a single global transformation (e.g. using a global affine transformation [187]) and for each point in Y

its correspondent point is given as the nearest neighbor point in X. Although we cannot guarantee finding a global minimum for problem (7.48), we are able to reduce the energy (7.30) at each iteration (given the matrices M, W), since the solution of each subproblem (7.51) reduces the total energy of the joint problem (s. e.g. [149]). In the praxis this is confirmed through the improvement of the matching accuracy (s. Fig. 7.5 (a) in the experimental section).

7.3.3 Choosing the Right Number of Clusters

In this section we describe how to automatically determine the complexity of the model, that is the number of affine transformations required for registration. The underlying idea is to measure the fluctuations in the registration results when random subsamples of the shapes are considered. For a given number of clusters k, our algorithm is run on b_{max} subsampled versions of the original shape Y (specifically, 60% of the points in the shape are randomly subsampled each time). Thus, we obtain the clustering results $\hat{M_b} \in \mathbb{R}^{n_s \times 1}$ ($b = 1, \cdots, b_{max}$, $n_s = \lfloor 0.6 * n \rfloor$), where $\hat{M_b}$ indicates the cluster number for each point in shape Y. Since the b_{max} clustering solutions are calculated on a subset of the points, they are extended to the whole shape using nearest neighbors for the missing points. The extended clustering solutions are referred to by $M_b \in \mathbb{R}^{n \times 1}$. Thereafter, pairwise distances between the different cluster solutions are calculated in order to evaluate the fluctuations in the results induced by the random subsampling. This is done using the minimal matching distance

$$\hat{d}_{\text{mmd}}(M_i, M_j) = \min_{\pi} \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[M_i(i) \neq \pi(M_j(i))]}, \tag{7.54}$$

where the minimum is taken over all permutations π of the k labels. In other words, $\hat{d}_{\text{mmd}}(M_i, M_j)$ measures the percentage of points which changed the assignment (up to a permutation). However, in order to avoid a bias when the number of clusters k is increased, $\hat{d}_{\text{mmd}}$ is normalized similar to [138] with the median $r(n)$ of pairwise distances between random labelings. Thus the fluctuations in the clustering results can be measured by:

$$d_{\text{mmd}}(M_i, M_j) := \frac{\hat{d}_{\text{mmd}}(M_i, M_j)}{r(n)} \tag{7.55}$$

In the case of stable clustering solutions, the pairwise distances $d_{\text{mmd}}(M_i, M_j)$ are expected to be near zero. In contrast, unstable solutions yield variations in the clusterings and large distances (s. Fig. 7.1 (d)). Therefore, we measure the instability of a solution by approximating the empirical distribution of pairwise distances $d_{\text{mmd}}(M_i, M_j)$ through a histogram $h \in \mathbb{R}^{\text{nbins} \times 1}$ over the distances, and define as a measure for the instability the sum of weighted counts:

$$\text{instab}(k) := \sum_{i}^{\text{nbins}} h(i) * c_h(i), \tag{7.56}$$

where $h(i)$ is the absolute count and $c_h(i)$ is the value of the histogram bin i. Since the number of runs b_{max} is the same for every value of k, the absolute counts of the histogram can be used without introducing any bias. This measure penalizes distances which are far from zero and thus, correspond to unstable clustering solutions for a certain value k.

Therefore, the ideal most stable number of affine transformations required for registration is defined as:

$$k_{\text{opt}} := \min_k \text{instab}(k). \tag{7.57}$$

Algorithm 7.1 Summary of the algorithm presented in this paper

```
     Input: original image $I_X$, reproduction $I_Y$
     Output: $k_{\text{opt}}$, $T^{\nu}_{\text{end}}$, $M_{\text{end}}$, $C_{\text{end}}$, $(\nu = 1, \cdots, k_{\text{opt}})$
 1   $\hat{X} \in \mathbb{R}^{n\times3}$, $\hat{Y} \in \mathbb{R}^{m\times3} \leftarrow$ Landmark points sampling
 2   $C_{\text{init}} \leftarrow$ Initial global affine registration
 3   for $k = 1, \cdots, k_{max}$   $\triangleright$ Number of groups
 4     for $b = 1, \cdots, b_{max}$ $\triangleright$ Iteration for subsamplings
 5       $X \in \mathbb{R}^{n_s\times3}$, $Y \in \mathbb{R}^{m_s\times3} \leftarrow$ Subsampling of $\hat{X}$, $\hat{Y}$
 6       $T_{\text{pool}} \leftarrow$ Initial pool of transformations
 7       do
 8         $M_{old} \leftarrow M^b,\ C_{old} \leftarrow C^b$
 9         $W^b \leftarrow \min_W E_{lin}(M, W, T^{\nu})$ $\triangleright$ Section 7.3.2
10         $M^b \leftarrow \min_M E_{lin}(M, W^b, T^{\nu})$
11         $T^{\nu}_b \leftarrow$ Weighted least squares given $M^b$, $W^b$
12         $C^b \leftarrow \min_C E_{lin} + E_{quad}$ $\triangleright$ Problem (7.51)
13       while$(C^b \neq C_{old} \wedge M^b \neq M_{old})$
14     end for
15     $\text{instab}(k) = \sum_i^{\text{nbins}} h^b(i) * c^b_h(i)$, $(b = 1, \cdots, b_{max})$$\triangleright$ Eq. (7.56)
16     $k \leftarrow k + 1$
17   end for
18   $k_{\text{opt}} \leftarrow \min_k \text{instab}(k)$ $\triangleright$ Eq. (7.57)
19   $M_{\text{end}}$, $C_{\text{end}}$, $T^{\nu}_{\text{end}} \leftarrow$ Repeat steps (6-13) once using $k \leftarrow k_{\text{opt}}$ and $X \leftarrow \hat{X}$, $Y \leftarrow \hat{Y}$
```

7.4 Experiments

7.4.1 Synthetic data

We first evaluate our algorithm on two frames of a synthetic image sequence. Fig. 7.1 (a) shows both frames in red and blue respectively. The head, both legs and tail were modified through affine transformations, and thus the global non-linear deformation between frames is known. In this case, around 4000 points are used to describe the shape and are uniformly sampled along the contours of the image. In order to carry out the stability analysis (s. Sec. 7.3.3) 60% of the points are uniformly subsampled and the algorithm is run $b_{max} = 60$ times for each given number of clusters k (on average the algorithm converged within 5 iterations each run). This resulted in 3600 pairwise distances for each k. This experiment is repeated 20 times, thus yielding the instability plot of Fig. 7.1 (c). The algorithm determined $k = 5$ to be the most stable number of groups. As Fig. 7.1 (b) shows (each color represents a single group), the corresponding groups are consistent with the manually introduced deformations. This experiment shows how our algorithm not only registers both shapes, but also how the inferred groups describe and visualize how the different local parts in the shape were truly deformed. Regarding this synthetic experiment, the distribution of the pairwise distances $d_{\text{mmd}}(M_i, M_j)$ (Eq. 7.55) for the

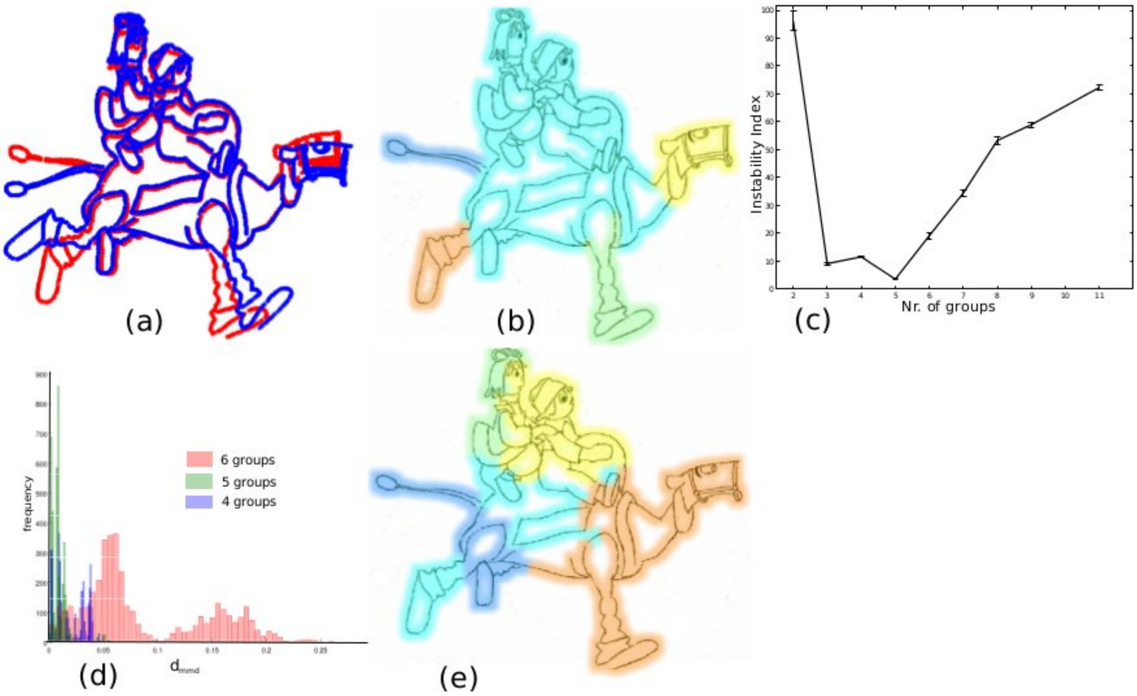


Figure 7.1: Results on Synthetic Data. (a) original image in blue and distorted image in red. (b) Groups found by our algorithm of Sec. 7.3.1 (each color corresponds to a different group). (c) Instability analysis for different numbers of groups. (d) Distribution of pairwise distances 7.54 for the most stable solutions. The distribution is not normal (e) Resulting clustering if the greedy single linkage algorithm of [181] is used for calculating the affine transformations instead of our LP-based method described in Sec. 7.3.2

most stable number of groups ($k = 4, 5, 6$) is also shown in Fig. 7.1 (d). The author in [241] (p. 5) mentions that a simple (normalized) mean over pairwise clustering distances $d_{\mathrm{mmd}}(M_i, M_j)$ is commonly used as instability measure. This methodology presuposes that the distribution of the pairwise distances is normal and thus the instability measure weights every pairwise distance equally. However, in Fig. 7.1 (d) we show that the distribution of pairwise distances is in general not normal. Therefore, our measure in Eq. **??** is more appropriate to describe the shape of the distribution since it weights the pairwise distances proportional to their occurrence. Finally, the benefit of our LP-based method (Sec. 7.3.2) for calculating the affine transformations and the assignment of points to them is evaluated by comparing our method with an alternative procedure based on the algorithm previously explored in [180]. Instead of using a pool of affine transformations T_{pool} and the LP-based method described in Sec. 7.3.2, we provided a single initial k-tuple of transformations by locally grouping points in a greedy manner based on their proximity and registration quality. This resulted in a defficient initialization which the sucessive updates of groups, transformations and correspondences could not correct. Whereas in Fig. 7.1 (e) we can observe how parts in the shape corresponding to different affine components wer mixed into the same group resulting in a clustering which is not consistent with the ground-truth, our current method (Fig. 7.1 (b)) groups the different shape parts correctly.

7.4.2 Reproductions of the Codex Manesse

In [181] we collected a corpus of 5 shapes coming from the Codex Manesse (reproduced between c. 1305 and c. 1340) and their corresponding reproductions commissioned in 1746/47 by J.J. Bodmer and J.J. Breitinger. Different regions of an image feature different transformations as can be seen from Fig. 7.3 and 7.4 where a single transformation does not suffice to bring the original and the reproduction into correspondence.

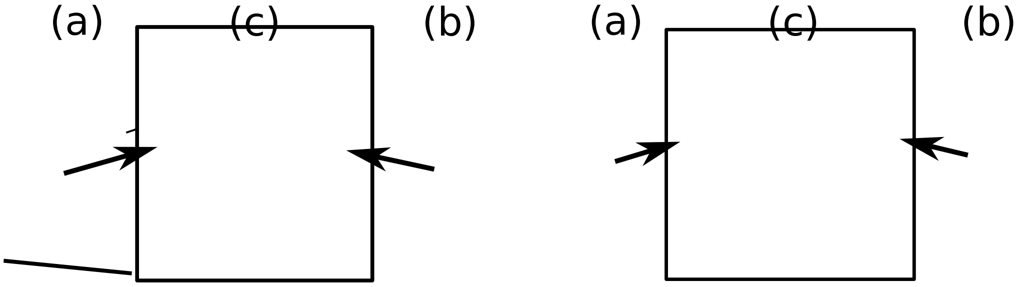


Figure 7.2: a) 14th century original image. b) 18th century manual reproduction. c) Transformation using a single affine transformation.

Only if the shape is decomposed into several affine transformations we achieve a consistent registration. An example of this is shown in Fig 7.4 (a) where both images are brought into alignment by means of piecewise affine transformations.

Since ground-truth for the correspondences between the shapes is known, it is possible to measure the registration quality of our method and compare it with other state-of-the-art algorithms. In Table 7.1 we show the mean squared error (MSE) of the registration for all shapes. The number of affine transformations for all models is automatically determined by our algorithm. Furthermore, we are also interested in measuring the ability of humans to perceive deformations in different parts of a shape. Therefore, we developed an interactive registration tool which was used by 5 experts to manually select the regions in the shape that according to their perception shared the same transformation. At the beginning of the experiment, both shapes were registered using a single affine transformation. Thereafter, each time a new group of points was selected the overall shape registration was updated enabling each user to see the result of his selection. Moreover, it was always possible to correct a group selected before. The average of the MSE over the 5 experts in the experiment is shown in table 7.1 under the row *human*. From the large MSE it becomes clear that the task of an art historian to manually analyze a shape to understand the drawing process is extremely difficult. Thus, a computer-based procedure is essential. The entries Kmeans and Ward in table 7.1 correspond to a piecewise affine registration based on the clustering of the displacement vectors between both shapes using Kmeans and Ward's method respectively. We have observed that clustering the error vectors featured only insufficient accuracy: contours have been distorted (e.g. stretched) and junctions are partly missing and thus affine deformations cannot be described by clustering the displacement term of the deformation. A similar method to Wang and Adelson [253] is also reimplemented (second row of Tab. 7.1). For this method not the displacement vectors but the parameters of the affine transformations contained in T_{pool} are directly clustered instead. Thereafter, we greedily iterate between the assignment of *each point* to

Figure 7.3: a) Transformation of an image using a piecewise affine transformation model.

the centroids (i.e. the affine transformation representing a group) based on its registration error and the refinement of the centroids themselves. The clustering of transformation parameters resulted in being unstable since they strongly varied depending on the locality of their support. Furthermore, the greedy assignment of points to transformations was also not optimal.

In table 7.1 we also add the output of the algorithm from [181] for comparison. In this case, we observed that areas in a shape were grouped based on their proximity due to the pairwise Euclidean distance term used in their objective function. This bias was also observed in the results of [63], where the assignments to transformations were also regularized by a Euclidean distance based term in their energy function. This fact had an important impact on the registration, since parts of the shape featuring different transformations were forced to be registered together and a bigger MSE was produced. Finally, it is important to remark that all of these methods with exception of the presented one only partially solve the full task since the correspondence between shapes is not calculated. Furthermore, it is not possible to automatically determine the model's complexity as we do in our method.

Since we have ground-truth for the correspondences we also measure the improvement of the matching quality between shapes induced by our algorithm. For this, we measure for each point y_i in shape Y the error produced between its estimated corresponding point $\sum_{j=1}^{m} C_{ij} x_j$ (as induced by the binary matrix C in our algorithm) and its true correspondent point x_i^{gt} (as provided in the ground-truth) in shape X. We then measure

Table 7.1: Reproductions of the Codex Manesse. Mean squared error (MSE) of the registration using ground-truth correspondences provided by [181]. The complexity of the piecewise affine transformation is automatically provided by our method.

	Registration quality (MSE)						
Shape ID (# groups)	[253]	**Kmeans**	**Ward**	[181]	**Human**	[63]	**Our Method**
shape 1 (10)	49.36	37.20± 2.25	35.46	34.71	57.91 ± 9.93	25.04	**24.89**
shape 2 (7)	109.26	80.98± 5.30	84.19	131.07	194.33± 6.34	260.60	**78.55**
shape 3 (6)	24.11	35.77± 1.27	36.15	45.62	37.06± 5.61	24.68	**21.41**
shape 4 (7)	28.57	37.37± 0.99	39.26	37.68	44.21 ± 7.97	35.77	**28.37**
shape 5 (4)	52.12	57.52± 4.83	52.66	66.89	60.23 ± 1.03	67.84	**45.86**
Average	52.68	49.76± 2.92	49.54	63.19	78.74 ± 6.17	82.78	**39.81**

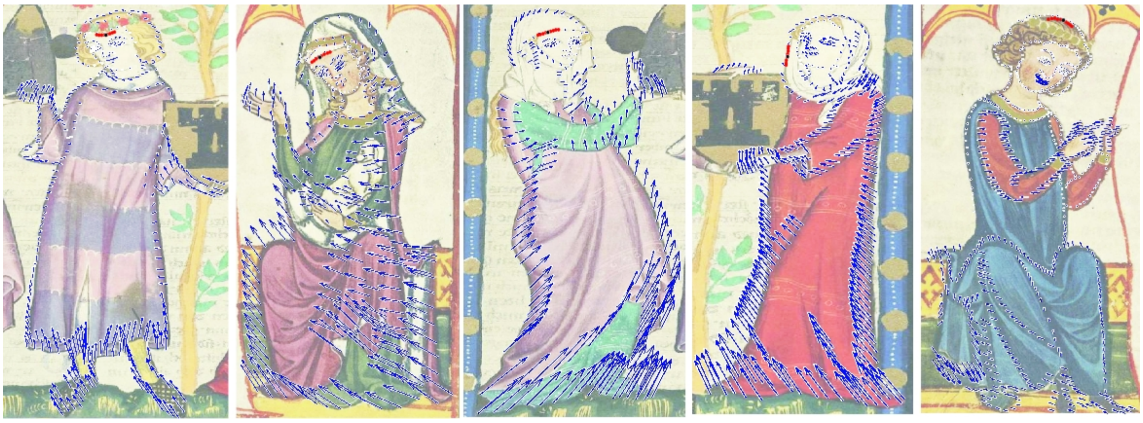


Figure 7.4: Inconsistency between parts. A single affine transformation is insufficient to model the distortion of the complete figure, since individual parts have been transformed differently by the artist. Each transformation is calculated using the points marked in red.

for a threshold on δ_{err} (which we then vary in turn) the percentage of points, where the estimated correspondences lie at most δ_{err} from the ground-truth. This yields a matching-accuracy curve depending on the parameter δ_{err}. In Fig. 7.5 (a) we show the relative improvement in the matching accuracy between the last and the first iteration (initialization) of our algorithm. Thus, optimizing the correspondence matrix C together with the groups M is beneficial for the registration process. Finally, we show in Fig. 7.5 (b) the stability of the solutions for the whole corpus of shapes and observe that a local minimum value indicating a stable solution for all shapes in the corpus is always obtained.

Finally, in order to compare the registration quality of our method with the CPD algorithm of [187] we utilize more complex medieval scenes. For this we use reproductions of the codex of Eike von Repgow's Sachsenspiegel ("Mirror of the Saxons") composed ca. 1220-1235 in eastern Saxony (s. Fig. 7.6 for one example). Whereas the CPD algorithm of [187] (using the default parameters) obtains a root MSE (RMSE) of 12.64 ± 11.96 for the registration error over 3 scenes, our method improves the registration with a RMSE of 8.79 ± 5.8. In contrast to this, registering the scene with a rigid transformation results in a poor RMSE of 20.65 ± 15.57. We observe that the improvement of our method over

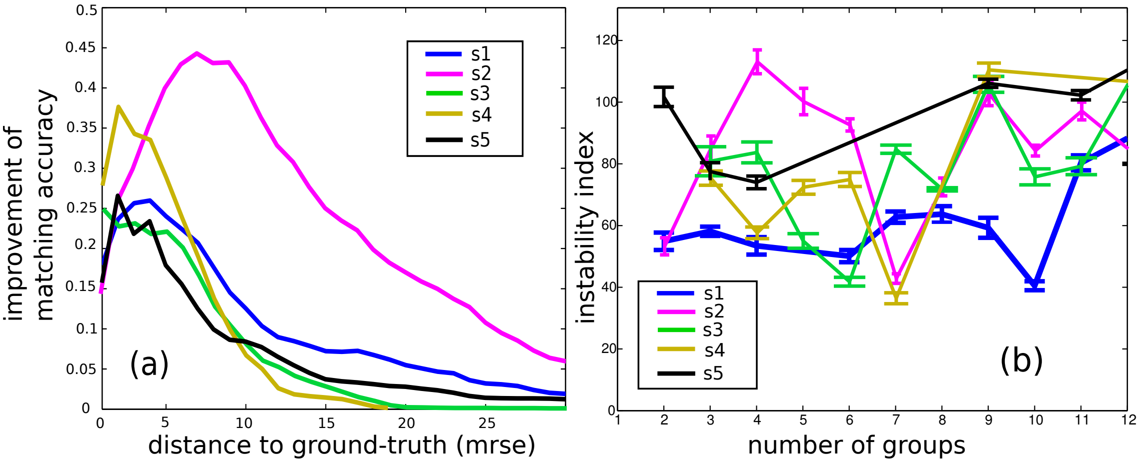


Figure 7.5: Results for the Codex Manesse corpus. (a) Improvement in the matching accuracy between the last and first iteration of our algorithm: we plot the difference between the matching accuracy curves of the last and the first iteration. Matching accuracy is the percentage of correspondences where the ground-truth correspondent point lies at most δ_{err} from the predicted correspondent point (s. Sec. 7.4.2) (b) Instability analysis for all shapes in the corpus and showing the standard deviation for each k.

the CPD algorithm is mainly due to the fact that whereas CPD regulates its complexity through a global parameter for the whole image, our method has a greater flexibility since it adapts its complexity, due to its piecewise nature, according to the underlying deformation.

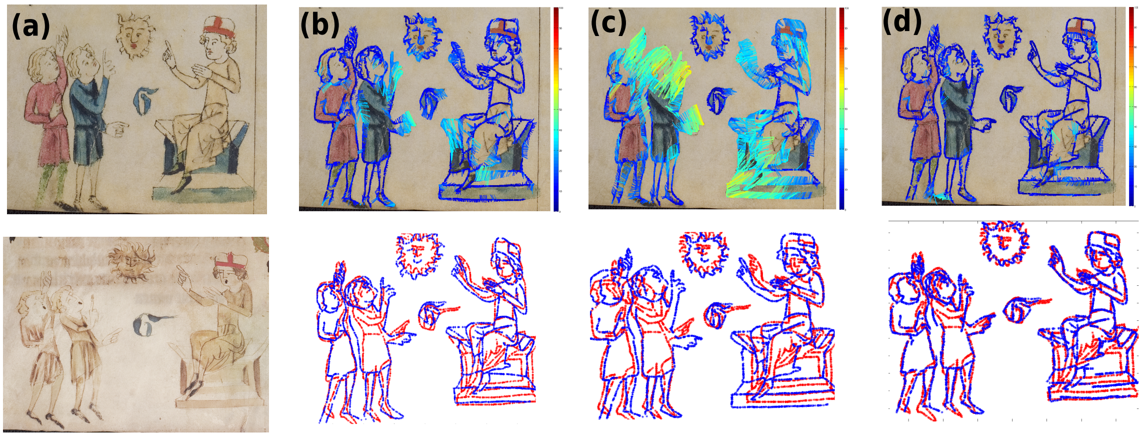


Figure 7.6: Registration quality for complex scenes. (a) reproductions of the codex of Eike von Repgowâs Sachsenspiegel (âMirror of the Saxonsâ) composed ca. 1220-1235 in eastern Saxony (b) Registration using the CPD algorithm of [187] (with a RMSE of 11.45 for this image). (c) Rigid registration (RMSE 17.48) (d) Registration results using our method (RMSE 7.78)

7.4.3 Michelangelo Reproductions

We also focused on the analysis of Michelangelo's ceiling fresco in the Sistine Chapel (1508-1512) and compare distinctive shapes with sketches, which were made by artists surrounding Michelangelo, probably after preparatory drawings or by Dutch artists after the original. The reason is that the differences between a drawing and a mural painting are greater than the tracings from book illustrations like the reproductions in the last section. Our aim here is not to reconsider the connoisseurs controversy about the attribution of these drawings, but to show how our automatic approach is used to analyze the reproduction process of an artwork which in turn is noteworthy for an art-historical analysis.

The first column in Fig. 7.8 shows the original fresco images. The second column shows two reproductions and a preparatory drawing. All three images in the second column seem to be reproduced exactly from the first column images. However, after applying an overall rigid transformation we see that the drawings feature important differences and show non-linear deviations from the fresco. This can be seen in the third row of figure 7.8, where the color of the arrows indicates the magnitude of the induced rigid registration error. Using our method it is possible to discover a structure in the overall deformation by observing the resulting groups obtained by our algorithm (s. Fig. 7.7). For instance, the Ignudo (i.e. the male nude flanking the Creation of Eve) in (a) features only two relevant deformations: whereas the upper and lower part of the body can be exactly registered to the other image, both parts together yield a non-linear deformation. From an artistic point of view this inconsistency can be explained by noting the difficulty of bringing both body parts into an appropriate distance and angle to each other by the artist during the reproduction of the fresco and alterations between these parts can easily be introduced in this procedure. Furthermore, the Prophet Jonah in (b) features very interesting groups: whereas the left leg fits using a single transformation, the right leg decomposes mainly into three groups which correspond to the observation that this body part substantially differs from the leg in the fresco. Furthermore, in (c) we can observe how the torso decomposes into the right and left arm indicating a deliberate amplification of the articulation in the sketching. Since our energy cost does not introduce any proximity-term which could bias the result, it can be concluded that the artist approached the reproduction by independently reproducing smaller parts corresponding to semantical entities. From an art historical point of view, whereas these parts can be considered as technically sensible, regions in the shape that were split in different groups indicate a possible difficulty of reproducing that area for the artist.

7.4.4 Implementation Details

For shape drawings we have to extract and deal with different contour thickness and texture. Hence, contours are extracted by convolving the image with Laplace of Gaussian (LoG) Filters of varying sigma ($\sigma = 0.8 + j * 0.4,\ j = 1, \dots, 9$) and then take the maximal response over all sigmas for every pixel. This kind of filter is suitable since it allows obtaining a single response for lines of varying thickness and ensures in praxis a good contrast between ridge response and background. Finally, non-maximum suppression followed by hysteresis thresholding is applied to obtain a single binary response. For images where shape is encoded through texture and color boundaries we use the Pb code ([163]) for edge extraction, which weights edge signals proportionally to their strength.

In Sec. 7.3.2 we have estimated correspondences for each group independently. When the group is too large (e.g. more than 1/5 of all points in the shape), each group is sub-

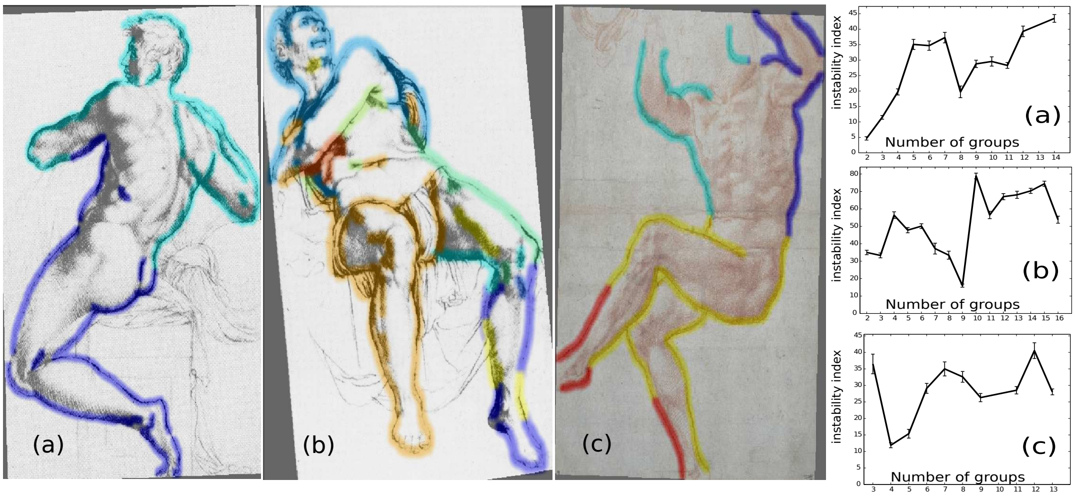


Figure 7.7: Analysis of the drawing process. First to third column: Different shapes were reproduced based in semantic entities (e.g. legs, arms, etc). The grouped parts are mostly anatomically or technically sensible whereas the parts that are split in different groups show a complex deformation for that area. Last column: corresponding instability analysis together with standard deviation

divided into smaller pieces based on a bottom-up contour grouping (using the Euclidean distance) and then the point correspondence for each subgroup is independently estimated. However, we force the groups to reach a minimum size to guarantee a robust matching.

7.5 Discussion

This paper has presented a novel approach for the analysis of alterations between artworks and their reproductions. Therefore, the overall shape deformation is represented by decomposition into a piecewise affine model. Model complexity was automatically estimated using a statistical stability analysis. The present contribution jointly estimated the correspondences between shapes, the affine structures in the shape, and the complexity required by the overall deformation model. We have tested our method in controlled scenarios, as well as with real historical images. Based on ground-truth correspondences between images from the Codex Manesse and their 18th century reproductions, we have observed an improvement over the state-of-the-art in both registration and matching quality. Furthermore, our algorithm outperformed a manual solution of the problem showing the benefit of this method for art historians. Finally, an important experimental finding was the discovery that the drawings of two of the Ignudi and the Prophet Jonah in the ceiling fresco of the Sistine Chapel featured different deformations. These deformations corresponded either to semantical entities of the shape (e.g. the arms in Fig. 7.7 (c)) or indicated slight modifications in the relative position of extremities (e.g. Fig. 7.7 (a) and (b)) by the artist.

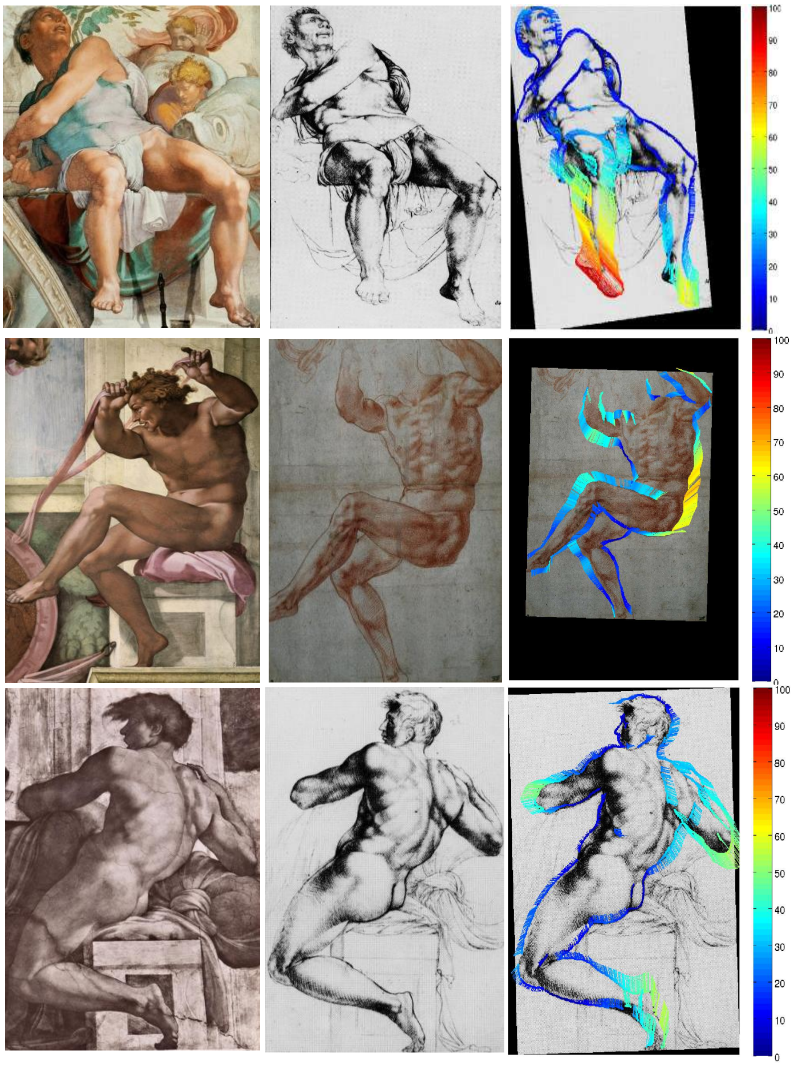


Figure 7.8: Parts of Michelangelo's ceiling fresco in the Sistine Chapel (1508-1512). First column: original fresco images. Second column: Sketches made after Michelangelo's own preparatory drawings or by Dutch artists after the original. Third column: Error between rigid registered images. The color of the arrows corresponds to the magnitude of the registration error.

CHAPTER 8

CONCLUSIONS

This thesis has dealt with the detection and classification of objects in visual images and with the analysis of shape changes between object instances. The driving force of this research was the idea that important properties for the automatic perception and understanding of objects are transmitted through their geometry or shape and thus, they should be exploited to solve the task at hand. The challenge consisted in using as little user supervision as possible.

Specifically, we investigated the usage of richer representations of shape in order to learn object models for recognition. The usage of curvature information in the underlying object representation provided quantitative evidence that it is possible to use this shape cue in a discriminative, robust, and reliable manner for object recognition. Our results showed that curvature information can be considered as orthogonal information to the state-of-the-art theme of histograms of oriented gradients for automatic visual search tasks. Combining both curvature and orientation of gradients, the accuracy and performance on standard datasets of a detection system solely based on HoG was significantly improved. Finally, it became clear that the proposed histogram-based curvature representation is generic, efficient to compute, and it is possible to effortlessly integrate it into all current histogram-based object models, thus granting a wide applicability. Despite the evidence that curvature is an important shape property for solving visual search tasks [261, 262] this cue had remained unused by state-of-the-art systems [80, 165, 245]. Therefore, it is remarkable that our work could bridge this gap for automatic object recognition.

This line of work also showed how to go beyond traditional bounding-box object representations for detection. This thesis introduced a method to learn object models while simultaneously learning to segregate objects from clutter and extract their overall shape without manual segmentation of the training samples. It was shown that it is possible to learn a prototypical set of segments and use it to represent and match objects of interest. Using the *Multiple Instance Learning* framework it was possible to capture the overall object shape in a model-driven manner by grouping the corresponding foreground regions of the query object and thus segregate the object from the background. The quantitative experimental results corroborated the main idea that segregating the shape of an object is relevant to increase the description power of the model and thus, improve the overall performance of the system. This is remarkable since using shape information within histogram-based object models has been typically either (i) avoided (e.g. [165, 104]), (ii) only a rough approximation of the object geometry has been used [80], or (iii) intensive supervision has been required ([99, 172, 247]). Only recent approaches like [37] have used shape for object recognition in a semi-supervised scenario. However, the specific assumptions made by those approaches (s. Sect. 4.2) have limited the performance of the system. Our results showed that it is possible to capture the overall object shape within a semi-supervised scenario by grouping corresponding foreground regions relying on machine learning techniques.

Another interesting evidence provided by this work is that it is possible to automatically detect objects in fully unsupervised scenarios making use of the property of perceiving shape equivalence. Due to the unsupervised scenario no annotation for the training data was available and thus, it was not possible to learn a statistical model for detecting and classifying objects. Shape equivalence refers to the ability of perceiving different object instances as sharing the same shape. Relevant objects within the historical dataset we analyzed were emphasized through annuli of light rays. This thesis explored the idea of considering the annuli as shape equivalent objects and devised a method for detecting them in an unsupervised manner. Thereafter, we were able to infer the size, position and scale of the emphasized object through the annuli detections. The task of detecting the rays of light and inferring the enclosed objects was not trivial, since finding objects of interest required finding the rays which surrounded them. However, recognizing which line segments in the image belonged to a ray annulus and which did not was related to the location of the query objects. The new application we introduced disclosed an automatic methodology to carry out iconographic analysis with the aim of revealing relevant meta-information about the images like the focus of attention of the artists, or more important the intention of those who commissioned those images.

Although object recognition is an important task in order to establish the functional meaning of an object by means of its classification, the current thesis also focused on the development of a method to detect and analyze the changes in shape between objects. This focus is remarkable since whereas object detection concentrates on learning the commonalities between object instances, analyzing the shape transformations focuses on describing the differences between them. Thus, both tasks complement each other and are necessary to better understand objects and develop automatic perception systems. Specifically, our method represented the overall complex deformation of an object using a piecewise linear model. It was possible to employ statistical stability analysis to estimate the model's complexity and thus, overcome one of the major limitations of state-of-the-art piecewise affine shape registration models (e.g. [48, 4, 188]): instead of manually selecting the regions in the shape that were affinely transformed, it was possible to formulate a joint optimization program which automatically identified each of these shape regions. Using this methodology it was possible to examine a novel interdisciplinary application for discovering and quantifying deformations induced during the reproduction process of artworks. At the same time, this novel application showed the fruitfulness of the interaction between Computer Vision and fields arising from the humanities. Specifically, chapters 5 and 7 showed that art history is able to pose interesting challenges to Computer Vision and motivate the development of new methodologies and approaches that bring the research field to its original goal: devise algorithms that enable computers to understand the visually perceivable world ([193]).

List of Publications

This dissertation has led to the following scientific articles

- Monroy, A., Bell, P., and Ommer, B. Morphological Analysis for Investigating Artistic Images. In Image and Vision Computing. Manuscript submitted for publication (2013).
- Monroy, A., and Ommer, B. Beyond Bounding-Boxes: Learning Object Shape by Model-driven Grouping In European Conference on Computer Vision (2012).
- Monroy, A., Bell, P., and Ommer, B. Shaping Art with Art: Morphological Analysis for Investigating Artistic Reproductions. In European Conference on Computer Vision VISART (2012)
- Monroy, A., Kroeger, T., Arnold, M., and Ommer, B. Parametric Object Detection for Iconographic Analysis. In Scientific Computing and Cultural Heritage (2011)
- Monroy, A., Eigenstetter, A., and Ommer, B. Beyond Straight Lines - Object Detection Using Curvature. In International Conference on Image Processing (2011)
- Monroy, A., Carque, B., and Ommer, B. Reconstructing the Drawing Process of Reproductions from Medieval Images. In International Conference on Image Processing (2011)

BIBLIOGRAPHY

[1] Alexe, B., Deselaers, T., and Ferrari, V. What is an object. In *CVPR* (2010).

[2] Andrews, S., Tscochantaridis, I., and Hofmann, T. Vector machines for multiple-instance learning. In *NIPS* (2003), vol. 15.

[3] Arbaláez, P., Maire, M., Fowlkes, C., and Malik, J. From contours to regions: An empirical evaluation. *CPVR:* (2009).

[4] Arsigny, V., Commowick, O., Pennec, X., and Ayache, N. A fast and Log-Euclidean polyaffine framework for locally affine registration. Tech. Rep. Resarch Report RR-5865, INRIA, 2006.

[5] Asada, H., and Brady, M. The curvature primal sketch. *PAMI: 8*, 1 (1986), 2–14.

[6] Awrangjeb, M., and Lu, G. Corner Detection based on the Chord-to-Point Distance Accumulation Technique. *Transactions on Multimedia 10*, 6 (2008), 1059–1072.

[7] Baca, M., Harping, P., Lanzi, E., McRae, L., and Whiteside, A. Cataloging Cultural Objects. A Guide to Describing Cultural Works and Their Images, 2006.

[8] Ballard, D. H. Generalizing the Hough Transform to Detect Arbirtrary Shapes. *Pattern Recognition 13*, 2 (1981), 111–122.

[9] Bao, S., Xiang, Y., and Savarese, S. Object Co-detection. *ECCV:* (2012).

[10] Bataille, A., and Dickinson, S. J. Coarse-to-Fine Object Recognition Using Shock Graphs. In *GbRPR* (2005).

[11] Bay, H., Ess, A., Tuytelaars, T., and van Gool, L. SURF: Speeded Up Robust Features. *CVIU: 110*, 3, 346–359.

[12] Bellmann, R. E. *Adaptive control processes: a guided tour.* Princeton University Press, 1962.

[13] Belongie, S., Malik, J., and Puzicha, J. Shape Matching and Object Recognition Using Shape Contexts. *PAMI 24*, 4 (2002), 509–522.

[14] Belyaev, A. Plane and space curves. curvature. curvature-based features. In *max-Planck-Institut fuer Informatik* (2004).

[15] Bengio, Y. Learning deep architectures for AI. *Foundations and Trends in Machine Learning 2*, 1 (2009), 1–127.

[16] Berg, A. C. *Shape Matching and Object Recognition.* PhD thesis, Computer Science Division, U.C. Berkeley, Dec. 2005.

[17] Berg, A. C., Berg, T. L., and Malik, J. Shape matching and object recognition using low distortion correspondence. In *CVPR* (2005), pp. 26–33.

[18] Berg, A. C., and Malik, J. Geometric blur for template matching. In *CVPR* (2001), pp. 607–614.

[19] Besl, P., and McKay, N. A Method for Registration of 3-D Shapes. *PAMI 14*, 2 (1992).

[20] Bezdek, J. *Pattern Recognition with fuzzy objective function algorithms.* Plenum Press, 1981.

[21] Bidermann, I., and Ju, G. Surface versus edge-based determinants of visual recognition. *Cognitive Psychology 1*, 20 (1988), 38–64.

[22] Blais, G., and Levine, M. Registering Multiview Range Data to Create 3D Computer Objects. *PAMI: 17*, 8 (1995).

[23] BonJour, L. *The Structure of Empirical Knowledge.* Harvard University Press, 1985.

[24] Bookstein, F. L. Biometrics, biomathematics and the morphometric synthesis. *Bulletin of Mathematical Biology 58*, 313–365.

[25] Bookstein, F. L. The Measurement of Biological Shape and Shape Change. *Lecture Notes on Biomathematics 24* (1978).

[26] Bookstein, F. L. Size and Shape Spaces for Landmark Data in Two Dimensions. *Statistical Science 1*, 2 (1986), 181–222.

[27] Bookstein, F. L. Principal warps: Thin-plate splines and the decomposition of deformations. *PAMI 11*, 6 (1989), 567–585.

[28] Borshukov, G., Bozdagi, G., Altunbasak, Y., and Tekalp, A. Motion segmentation by multi-stage affine classification. *IEEE Trans. on IP 6*, 11 (1997).

[29] Bosch, A., Zisserman, A., and Muoz, X. Scene classification using a hybrid generative/discriminative approach. *PAMI: 30*, 04 (2008), 712–727.

[30] Boykov, Y., Veksler, O., and Zabih, R. Fast Approximate Energy Minimization Via Graph Cuts. *PAMI:*, 29 (2001), 1222–1239.

[31] Brown, J. M., Enns, J. T., and May, J. G. Visual Search for simple volumetric shapes. *Perception and Psychophysics 51*, 1 (1992), 40–48.

[32] Bunescu, R. C., and Mooney, R. J. Multiple instance learning for sparse positive bags. . In *ICML:* (2007).

[33] Burkard, R., Dell'Amico, M., and Martello, S. *Assignment Problems.* SIAM, 2009.

[34] Burl, M. C., Weber, M., and Perona, P. A Probabilistic Approach to Object Recognition Using Local Photometry and Global Geometry. In *ECCV* (1998), pp. 628–641.

[35] Canny, J. A Computational Approach to Edge Detection. *PAMI 8*, 6 (1986), 679–698.

[36] Carqué, B. Zur manuellen Reproduktion der Miniaturen. *Der Codex Manesse und die Entdeckung der Liebe* (2010).

[37] Carreira, J., Li, F., and Sminchisescu, C. Object Recognition by Sequential Figure-Ground Ranking. *IJCV* (November 2011).

[38] Carreira, J., and Scminchisescu, C. Constrained Parametric Min-Cuts for Automatic Object Segmentation. In *CVPR* (2010).

[39] Chang, Y. S., and Stork, D. G. Warping realist art to ensure consitent perspective: A new software tool for art investigations. In *Human vision and electronic imaging* (2012).

[40] Chen, Y., and Medioni, G. Object modeling by Registration of Multiple Range Images. In *In ICRA* (1991).

[41] Chen, Y., Zhu, L., and Yuille, A. Active Mask Hierarchies for Object Detection. In *ECCV* (2010).

[42] Cheverud, J. M., Lewis, J. L., Bachrach, W., and Lew, W. D. The Measurement of form and variation in form: an application of three dimensional quantitative morphology by finite element methods. *Journal of Physical Anthropology 62*, 151–165.

[43] Chui, H., and Rangarajan, A. A feature Registration Framwork Using Mixture Models. In *Proc. IEEE Workshop Math. Methods in biomedical Image Analysis* (2000), pp. 190–197.

[44] Chui, H., and Rangarajan, A. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding 89*, 2-3 (2003), 114–141.

[45] Coeurjolly, D., Serge, M., and Laure, T. Discrete curvature based on osculating circles estimation. *Lecture Notes in Computer Science* (2001).

[46] Cohen, E. H., Barenholtz, E., Singh, M., and Feldman, J. What Change Detection Tells us About the Visual Representation of Shape. *Journal of Vision*, 5 (2005), 313–321.

[47] Colonder, M., Lepetit, V., Strecha, C., and Fua, P. BRIEF: Binary Robust Independent Elementary Features . In *ECCV:* (2010), pp. 778–792.

[48] Commowick, O., Arsigny, V., Isambert, A., Costa, J., Dhermain, F., Bidault, F., Bondiau, P.-Y., Ayache, N., and Malandain, G. An Efficient Locally Affine Framework for the Smooth Registration of Anatomical Structures. *Medical Image Analysis 12*, 4 (2008), 427–441.

[49] Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. Training models of shape from sets of examples. In *Im BMVC* (1992), pp. 9–18.

[50] Coreth, E., and Schöndorf, H. *Philosophie des 17. und 18. Jahrhunderts*, 4. auflage ed. Kohlhammer Urban Taschenbücher, 2008.

[51] Cortes, C., and v. Vapnik. Support-vector networks. vol. 20, pp. 273–297.

[52] Cour, T., Srinivasan, P., and Shi, J. Balanced graph matching.

[53] Crandall, D. J., Felzenszwalb, P. F., and Huttenlocher, D. P. Spatial Priors for Part-Based Recognition Using Statistical Models. In *CVPR* (2005), pp. 10–17.

[54] Cristinacce, D., and Cootes, T. Boosted regression active shape models. In *BMVC* (2007).

[55] Cross, A. D., and Hancock, E. R. SGraph Matching with Dual Step EM Algorithm. *PAMI: 20*, 1 (1998), 1236–1253.

[56] Csurka, G., Dance, C. R., Fan, L., Willamowski, J., and Bray, C. Visual categorization with bags of keypoints. In *ECCV* (2004), *Workshop on Statistical Learning in Computer Vision.*

[57] Csurka, G., Dance, C. R., Fan, L., Willamowski, J., and Bray, C. Visual categorization with bags of keypoints. In *ECCV* (2004), *Workshop Stat. Learn. in Comp. Vis.*

[58] da Silva, N., and Costeira, J. Subspace segmentation with outliers: A grassmannian approach to the maximum consesus subspace. *CVPR:* (2008).

[59] d'Aische, A. d. B., de Craene, M., Macq, B., and Warfield, W. An articulated registration method. In *In ICIP:* (2005).

[60] Dalal, N., and Triggs, B. Histograms of oriented gradients for human detection. In *CVPR* (2005).

[61] Dalal, N., and Triggs, B. Histograms of oriented gradients for human detection. In *CVPR* (2005).

[62] Debus, A. G. *Man and nature in the Renaissance.* Cambridge University Press.

[63] Delong, A., Osokin, A., Isack, H., Boykov, Y., et al. Fast Approximate Energy Minimization with Label Costs. *CVPR:* (2010).

[64] Deng, J., Krause, J., and Fei-Fei, L. Fine-Grained Crowdsourcing for Fine-Grained Recognition. In *In CVPR* (2013).

[65] Desai, C., Ramanan, D., and Fowlkes, C. Discriminative models for mulit-class object layout. In *ICCV* (2009), pp. 229–236.

[66] Deselaers, T., and Ferrari, V. A conditional random field for multiple-instance learning . In *ICML:* (2010).

[67] Dietrich, T. G., Lathrop, R. H., and Lozano-Pérez, T. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence 89* (1997), 31.71.

[68] Donoser, M., Riemenschneider, H., and Bischof, H. Efficient Partial Shape Matching of Outer Contours. In *ACCV:* (2009), pp. 281–292.

[69] Drucker, H., Kaufman, J. C., Smola, A. J., and Vapnik, V. N. Support Vector Regression Machines. In *NIPS:* (1996), vol. 9, MIT Press, pp. 155–161.

[70] Dryden, I. L., and Mardia, K. V. *Statistical Shape Analysis.* Wiley series in probability and statistics, 1998.

[71] Duchenne, O., Bach, F., Kweon, I., and Ponce, J. A Tensor-based algorithm for high-order graph matching. *CVPR:* (2009).

[72] Duchon, J. Interpolation des fonctions de deux variables suivant la principe de la flexion des plaques minces. *RAIRO Anal. Numé. 10* (1976), 5–12.

[73] Duda, R. O., and Hart, P. E. Use of the Hough Transformation to Detect Lines and Curves in Pictures. *Comm. ACM 15* (1972), 11–15.

[74] Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification*, 2nd ed. John Wiley, New York, NY, 2001.

[75] Dürer, A. *Vier Bücher von Menschlicher Proportion.* 1528.

[76] Eggert, D., Lorusso, A., and fisher, R. B. Estimating 3D Rigid Body Transformations. *MVA 9*, 5/6 (1997).

[77] Eigenstetter, A., and Ommer, B. Visual Recognition using Embedded Feature Selection for Curvature Self-Similarity. In *NIPS:* (2012).

[78] Elhamifar, E., and Vidal, R. Sparse subspace clustering. *CVPR* (2009).

[79] Estrozi, L. F., Filho, L. G., Bianchi, G. C., Jr., R. C., and Costa, L. d. F. 1D and 2D fourier-based approaches to numeric curvature estimation and their comparative performance assessment. In *Digital Signal Processing* (2003).

[80] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. Object Detection with Discriminatively Trained Part-Based Models. *PAMI 32*, 9 (sept. 2010), 1627–1645.

[81] Felzenszwalb, P. F. Representation and Detection of Deformable Shapes. *PAMI 27*, 2 (2005), 208–220.

[82] Felzenszwalb, P. F., and Huttenlocher, D. P. Efficient Matching of Pictorial Structures. In *CVPR* (2000), pp. 66–73.

[83] Felzenszwalb, P. F., and Huttenlocher, D. P. Pictorial Structures for Object Recognition. *IJCV 61*, 1 (2005).

[84] Fergus, R., Fei-Fei, L., Perona, P., and Zisserman, A. Learning Object Categories from Google's Image Search. In *ICCV* (2005), pp. 1816–1823.

[85] Fergus, R., Perona, P., and Zisserman, A. Object Class Recognition by Unsupervised Scale-Invariant Learning. In *CVPR* (2003), pp. 264–271.

[86] Fergus, R., Perona, P., and Zisserman, A. A Visual Category Filter for Google Images. In *ECCV* (2004), pp. 242–256.

[87] Fergus, R., Perona, P., and Zisserman, A. A Sparse Object Category Model for Efficient Learning and Exhaustive Recognition. In *CVPR* (2005), pp. 380–397.

[88] Ferrari, V., Fevrier, L., Jurie, F., and Schmid, C. Groups of Adjacent Contour Segments for Object Detection. In *PAMI:* (2008).

[89] Ferrari, V., Fevrier, L., Jurie, F., and Schmid, C. Groups of adjacent contour segments for object detection. *PAMI 30*, 1 (2008), 36–51.

[90] Ferrari, V., Jurie, F., and Schmid, C. Accurate Object Detection with Deformable Shape Models Learnt from Images. In *CVPR* (2007).

[91] Ferrari, V., Jurie, F., and Schmid, C. From images to shape models for object detection. Tech. Rep. 6600, INRIA, Grenoble, 2008.

[92] Fidler, S., and Leonardis, A. Towards scalable representations of object categories: Learning a hierarchy of parts . In *CVPR:* (2007).

[93] Fischler, M. A., and Elschlager, R. A. The Representation and Matching of Pictorial Structures. *IEEE Transactions on Computers c-22*, 1 (1973), 67–92.

[94] Forsyth, D. A., and Ponce, J. *Computer Vision: A Modern Approach.* Prentice Hall, Upper Saddle River, NJ, 2003.

[95] Foster, D. H., and Westland, S. Categorical and noncategorical discrimination of curved lines depends on stimulus duration, not performance level. In *Perception* (1989).

[96] Freeman, H., and Davis, L. S. A corner finding algorithm for chain code curves. In *Trans. on Computers* (1977), vol. 29, pp. 297–303.

[97] Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics 36*, 4 (1980), 193–202.

[98] Gall, J., and Lempitsky, V. Class-specific hough forests for objct detection. *CPVR:* (2009).

[99] Gao, T., Packer, B., and Koller, D. A Segmentation-aware Object Detection Model with Occlusion Handling. In *CVPR* (2011), pp. 1361–1368.

[100] Gehler, P. V., and Chapelle, O. Deterministic annealing for multiple-instance learning. In *In AISTATS* (2007).

[101] Gold, S., and Rangarajan, A. A graduated assignment algorithm for graph matching. *PAMI: 18*, 4, 377–388.

[102] Grant, E. *A History of Natural Philosophy: From the Ancient World to the Nineteenth Century.* Cambridge University Press, 2007.

[103] Grauman, K., and Darrell, T. The pyramid match kernel: Discriminative classification with sets of image features. In *ICCV* (2005).

[104] Grauman, K., and Darrell, T. Pyramid Match Kernels: Discriminative Classification with Sets of Image Features. Tech. Rep. MIT-CSAIL-TR-2006-020, 2006.

[105] Grimson, W., and Huttenlocher, D. On the sensitivity of the hough transform for object recognition. *PAMI 12*, 3 (1990), 255–274.

[106] Gu, C., Lim, J., Arbeláez, J., and Malik, J. Recognition using Regions. In *ICCV* (2009).

[107] Gu, C., Lim, J. J., Arbeláez, P., and Malik, J. Recognition using Regions. *CVPR* (2009).

[108] Gumhold, S. Designing optimal curves in 2d. In *CEIG* (2004).

[109] Han, J., and Poston, T. Chord-to-point distance accumulation and planar curvature: a new approach to discrete curvature. *Pattern Recogn. Lett. 22*, 10 (2001).

[110] Harzallah, H., Jurie, F., and Schmid, C. Combining efficient object localization and image classification. In *In ICCV* (2009).

[111] He, X., and Yung, N. Corner detector based on global and local curvature properties. In *Optical Engineering* (2008), vol. 47.

[112] Heimsoeth, H. *Die Sechs Großen Themen der Abendläandischen Metaphyik und der Ausgang des Mittelalters.* WBG, 1974.

[113] Hinton, G. E., Williams, C. K. I., and Revow, M. D. Adaptive Elastic Models for Hand-Printed Character Recognition. pp. 512–519.

[114] Ho, J., Yang, M.-H., Lim, J., Lee, K.-C., and Kriegman, D. Clustering appearances of objects under varying illumination conditions. *CVPR:* (2003).

[115] Hobbes, T. *Leviathan.* 1651.

[116] Hongsheng, L., Huang, J., Zhang, S., and Huang, X. Optimal Object Matching via Convexification and Composition. In *ICCV* (2011).

[117] Hough, P. Machine analysis of bubble chamber pictures. In *Proc. Int. Conf. High Energy Accelerators and Instrumentation* (1959).

[118] Hubel, D. H., and Wiesel, T. N. Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex. *Journal of Physiology 160* (1962), 106–154.

[119] Hurley, J. R., and Cattell, R. B. The Procrustes program: producing direct rotation to test a hypothesied factor structure. *Behavioural Science 42*, 7 (1962), 258–262.

[120] J., S., Blake, A., and Cipolla, R. Multi-scale categorical object recognition using contour fragments. . *PAMI 30*, 7 (2007), 1270–1281.

[121] Jähne, B. *Digitale Bildverarbeitung*, 6. auflage ed. 2005.

[122] jain, A., Duin, R., and Mao, J. Statistical pattern recognition: A review. *PAMI: 22* (2000), 4–37.

[123] Jain, A. K., Zhong, Y., and Lakshmanan, S. Object matching using deformable templates. *PAMI 18*, 3 (1996), 267–278.

[124] Joshi, A., and Lee, C. H. On the Problem of Correspondence in Range Data and Some Inelastic Uses for Elastic Nets. *IJCV: 6*, 3 (1995), 716–723.

[125] J.Tippet, et al. Machine perception of three-dimensional solids. In *optical and Electro-Optical Information Processing* (1965).

[126] Kanatani, I. Motion segmentation by subspace separation and model selection. *ICCV:* (2001).

[127] Kass, M., Witkin, A., and Terzopoulos, D. Snakes: Active contour models. *ICCV 1*, 4 (1987), 259–268.

[128] Kaufman, L., and Rouseeuw, P. *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley, 1990.

[129] Kendall, D. Shape Manifolds, Procrustean Metrics and Complex Projective Spaces. *Bulletin of the London Mathematical Society 16*, 2 (1984), 81–121.

[130] Kendall, D. G. The diffusion of shape . *Advances in Applied Probability* (1977), 428–430.

[131] Kent, J. T. The complex bingham distribution and shape analysis. *Journal of the Royal Statistical Society, Series B 56*, 285–299.

[132] Kent, J. T., and Mardia, K. V. *Statistical Shape methodology.* 1994, pp. 443–452.

[133] Komodakis, N., Paragios, N., and Tziritas, G. Clustering via LP-based Stabilities. In *NIPS:* (2009).

[134] Krige, D. G. A Statistical approach ot some basic mine valuation problens on the Wiwatersrand. *Journal of the Chem., Metal. and Mining Soc. 14*, 1 (1951), 119–139.

[135] Krizhevsky, A., Sutskever, I., and Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS:* (2012), pp. 1106–1114.

[136] Lades, M., Vorbrüggen, J. C., Buhmann, J. M., Lange, J., von der Malsburg, C., Würtz, R. P., and Konen, W. Distortion Invariant Object Recognition in the Dynamic Link Architecture. *IEEE Transactions on Computers 42* (1993), 300–311.

[137] Lampert, C. H., Blaschko, M. B., and Hofmann, T. Beyond sliding windows: Object localization by efficient subwindow search. In *CVPR* (2008).

[138] Lange, T., Roth, V., Braun, M. L., and Buhmann, J. M. Stability-Based Validation of Clustering Solutions. *Neural Computation 16*, 6 (2004), 1299–1323.

[139] Lauer, F., and Schnorr, C. Spectral clustering of linear subspaces for motion segmentation. *ICCV* (2009).

[140] Lazebnik, S., Schmid, C., and Ponce, J. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *CVPR* (2006).

[141] Lazic, N., Givoni, I., and Frey, B. FLoSS: Facility Location for Subspace Segmentation. *ICCV:* (2009).

[142] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE 86*, 11 (1998), 2278–2324.

[143] Leibe, B. *Interleaved Object Categorization and Segmentation.* PhD thesis, ETH Zurich, Oct. 2004.

[144] Leibe, B., Leonardis, A., and Schiele, B. Combined Object Categorization and Segmentation with an Implicit Shape Model. In *ECCV* (2004), *Workshop Stat. Learn. in Comp. Vis.*

[145] Leibe, B., Leonardis, A., and Schiele, B. Combined Object Categorization and Segmentation with an Implicit Shape Model. In *ECCV* (2004), *Workshop on Statistical Learning in Computer Vision.*

[146] Leibe, B., Leonardis, A., and Schiele, B. Robust object detection with interleaved categorization and segmentation. *IJCV 77*, 1-3 (2008), 259–289.

[147] Leibe, B., and Schiele, B. Scale-Invariant Object Categorization using a Scale-Adaptive Mean-Shift Search. In *Pattern Recognition (Symposium of the DAGM)* (2004), vol. 3175 of *LNCS*, pp. 145–153.

[148] Leordeanu, M., and Herbert, M. A Spectral technique for correspondence problems using pairwise constraints. In *ICCV* (2005).

[149] Leordeanu, M., Herbert, M., and Sukthankar, R. An Integer Projected Fixed Point Method for Graph Matching and MAP Inference . In *NIPS:* (2009), Springer.

[150] Leordeanu, M., Sukthankar, R., and Hebert, M. Unsupervised Learning for Graph Matching. *IJCV 96* (2012), 28–45.

[151] Levin, A., and Weiss, Y. Learning to combine bottom-up and top-down segmentation. *IJCV*, 81(1) (2009), 105–118.

[152] Lew, W. D., and Lewis, J. L. An anthropometric scaling method with application to the kneww joint. 171–184.

[153] Lewiner, T., Jr., J. D. G., Lopes, H., and Craizer, M. Arc-length based curvature estimator. In *SIBGRAPI* (2004).

[154] Liu, M., Tuzel, O., Veeraghavan, A., and Chellappa, R. Fast directional chamfer matching. In *CVPR* (2010).

[155] Locke, J. Essay concerning Humane Understanding, 1690.

[156] Lowe, D. Object recognition from local scale-invariant features. In *ICCV* (1999), pp. 1150–1157.

[157] Lowe, D. G. The viewpoint consistency constraint. *IJCV 1*, 1 (1987), 57–72.

[158] Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV 60*, 2 (2004), 91–110.

[159] Luo, B., and Hancock, E. R. Structural Graph Matching Using the EM Algorithm and Singular Value Decomposition. *PAMI: 23*, 10 (2001), 1120–1136.

[160] Luo, B., and Hancock, E. R. A Unified Framework for Alignment and Correspondence. *CVIU:* (2006), 937–940.

[161] Ma, T., Yang, X., and Latecki, L. From partial shape matching through local deformation to robust global shape similarity for object detection .

[162] Macrini, D., Shokoufandeh, A., Dickinson, S., Siddiqi, K., and Zucker, S. View-based 3-D object recognition using shock graphs. *CVPR:* (2002).

[163] Maire, M., Arbelaez, P., Fowlkes, C., and Malik, J. Using contours to detect and localize junctions in natural images. In *CVPR* (2008).

[164] Maji, S., Berg, A. C., and Malik, J. Classification using Intersection Kernel Support Vector Machines is Efficient. In *CVPR* (2008).

[165] Maji, S., and Malik, J. Object detection using a max-margin hough transform. In *CVPR* (2009).

[166] Malisiewicz, T., and Efros, A. Improving spacial support for objects via multiple segmentations. In *BMVC* (2007).

[167] Mark, E., Gool, L., Williams, C., Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2007 (VOC2007). In *Results* (2007).

[168] Marr, D. Artificial intelligence: a personal view. In *Mind Design*. MIT Press, ch. 4, pp. 129–142.

[169] Marr, D. Approaches to biological information processing. . vol. 190, pp. 875–876.

[170] Marr, D. *Vision*. W. H. Freeman, San Francisco, CA, 1982.

[171] Marr, D., and Hildereth, E. Theory of edge detection. *Proc. of the Royal Society of London 207*, 1167 (1980), 187–217.

[172] Marszalek, M., and Schmidt, C. Accurate Object Recognition with Shape Masks. *IJCV*, 97 (2011), 191–209.

[173] Martin, D., Fowlkes, C., and Malik, J. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI 26*, 5 (2004).

[174] Masuda, T., Sakaue, K., and Yokoya, N. Registration and Integration of Multiple Range Images for 3-D Model Construction. In *In ICPR:* (1996).

[175] McNeill, G., and Vijayakumar, S. A Probabilistic Approach to Robust Shape Matching. *ICIP:* (2006), 937–940.

[176] Medioni, G., and Yasumoto, Y. Corner detection and curve representation using cubic b-splines. *Computer Vision, Graphics and Image Processing 39*, 3 (1987), 267–278.

[177] Meinguet, J. Multivariate interpolation at arbitrary points made simple. *Z. Angewandte Math. Phys. 30* (1979), 292–304.

[178] Mikolajczyk, K., and Schmid, C. A Performance Evaluation of Local Descriptors. *PAMI: 10*, 27, 1615–1630.

[179] Mockhtarian, F., and Mackworth, A. Scale-based description and recognition of planar curves and two-dimensional shapes. In *PAMI:* (1986), vol. 8, pp. 34–43.

[180] Monroy, A., Bell, P., and Ommer, B. Shaping Art with Art: Morphological Analysis for Investigating Artistic Reproductions. In *ECCV (VISART)* (2012), Springer.

[181] Monroy, A., Carque, B., and Ommer, B. Reconstructing the Drawing Process of Reproductions from Medieval Images. In *ICIP* (2011).

[182] Monroy, A., Eigenstetter, A., and Ommer, B. Beyond Straight Lines - Object Detection Using Curvature. In *ICIP* (2011).

[183] Monroy, A., Kroeger, T., Arnold, M., and Ommer, B. Parametric Object Detection for Iconographic Analysis. In *Scientific Computing and Cultural Heritage* (2011).

[184] Monroy, A., and Ommer, B. Beyond Bounding-Boxes: Learning Object Shape by Model-driven Grouping. *ECCV:* (2012).

[185] Monroy, A., and Ommer, B. Morphological Analysis for Investigating Artistic Images. *Image and Vision Computing* (Manuscript submitted for publication).

[186] Moss, M. L., Vilman, H., Moss-Saletijn, L., Sen, K., Pucciarelli, H. M., and Skalak, R. Studies on rothocephalization: Growth behaivour of the rat skull in the period 13-19 days as described by the finite element method. *American Journal of Physical Anthropology 72* (1987), 323–342.

[187] Myronenko, A., and Song, X. Point Set Registration:Coherent Point Drift. *PAMI: 32*, 12 (2010), 2262–2275.

[188] Narayanan, R., Fessler, J. A., Park, H., and Meyer, C. R. Diffeormorphic nonlinear transformations: A local parametric approach for image registration. In *In IPMI* (2005), pp. 174–185.

[189] Ng, A. Y., and Jordan, M. I. On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes. In *Proceedings of the Conference on Advances in Neural Information Processing Systems* (2002), no. 14, pp. 841–848.

[190] Odone, F., Barla, A., and Verri, A. Building kernels from binary strings for image matching. *Transac. on Image Processing 14*, 2 (2005), 349–361.

[191] Ojala, T., Pietikainen, T., and Harwood, D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In *ICPR* (1994), vol. 1, pp. 582–585.

[192] Okabe, A., Boots, B., and Sugihara, K. *Spatial Tesselations.* Wiley, 1992.

[193] Ommer, B. *Seeing the Objects Behind the Parts: Learning Compositional Models for Visual Recognition.* VDM Verlag, 2008.

[194] Ommer, B., and Malik, J. Multi-Scale Object Detection by Clustering Lines. In *ICCV* (2009).

[195] Opelt, A., Pinz, A., and Zisserman, A. Incremental learning of object detectors using a visual shape alphabet. In *CVPR* (2006).

[196] Palmer, S. E. *Vision Science:Photons to Phenomenology.* MIT Press, Cambridge, MA, 1999.

[197] Papademetris, X., Dione, D. P., Dobrucki, L. W., Staib, L. H., and Sinusas, A. J. Articulated rigid registration for serial lower-limb mouse imaging. In *In MICCAI* (2005), pp. 919–926.

[198] Pedersoli, M., Vedaldi, A., and Gonzalez, J. A coarse-to-fine approach for fast deformable object detection. In *CVPR* (2011).

[199] Pitiot, A., Bardinet, E., Thompson, P. M., and Malandain, G. Piecewise Affine Registration of Biological Images for volume reconstruction. *Medical Image Analysis 10*, 3 (2006), 465–483.

[200] Plato. *The Republic.* No. 514a-520b in Book VII.

[201] Plato. *Timaios.* No. 29b,37c.

[202] Poggio, T., and Edelman, S. A network that learns to recognize 3d objects. *Nature:*, 343 (1990), 263–266.

[203] Pontil, M., Rogai, S., and Verri, A. Recognizing 3-D Objects with Linear Support Vector Machines. In *ECCV* (1998), pp. 469–483.

[204] Pratt, V. Direct least-squares fitting of algebraic surfaces. In *Computer & Graphics* (1987), vol. 21, pp. 145–152.

[205] Pulli, K. Multiview Registration for Large Data Sets.

[206] Rao, C., et al. *Linear Models: Least Squares and Alternatives.* Springer Series in Statistics. 1999.

[207] Razavi, N., NeuerAutor2, Gall, J., and van Gool, L. Scalable mulit-class object detection. In *CVPR* (2011).

[208] Revow, M., Williams, C. K. I., and Hinton, G. E. Using Generative Models for Handwritten Digit Recognition. *PAMI: 18*, 6 (1996), 592–606.

[209] Ricken, F. *Philosophie der Antike*, 6. auflage ed. Kohlhammer Urban Taschenbücher, 2007.

[210] Riemenschneider, H., Donoser, H., and Bischof, H. Using partial edge contour matches for efficient object category localization. . In *ECCV:* (2010).

[211] Riesenhuber, M., and Poggio, T. Models of object recognition. *Nature Neuroscience 3*, 11 (2000), 1199–1204.

[212] Roberts, L. G. *Machine Perception Of Three-Dimensional Solids.* PhD thesis, Massachusetts Institute of Technology, 1963.

[213] Rublee, E., Rabaud, V., Konolige, K., and Gary, B. ORB: An Efficient Alternative to SIFT or SURF. *ICCV:* (2011).

[214] Rusinkiewicz, S., and Levoy, M. Efficient Variants of the ICP Algorithm. In *3DIM* (2001).

[215] Sala, P., and Dickinson, S. Contour grouping and abstraction using simple part models. In *ECCV:* (2010).

[216] Schellewald, C., Roth, S., and Schnörr, C. Performance evaluation of a Convex Relaxation Approach to the Quadratic Assignment of Relational Object Views. *IVCJ 25*, 8 (2007).

[217] Schellewald, C., and Schnorr, C. Probabilistic subgraph matching based on convex relaxation. In *EMMCVPR:* (2005).

[218] Schneiderman, H. Learning a Restricted Bayesian Network for Object Detection. In *CVPR* (2004), pp. 639–646.

[219] Schneiderman, H., and Kanade, T. A Statistical Method for 3D Object Detection Applied to Faces and Cars. In *CVPR* (2000), pp. 1746–1759.

[220] Schnitzpan, P., Fritz, M., Roth, S., and Schiele, B. Discriminative structure learning of hierarchical representations for object detection. In *CVPR* (2009), pp. 2238–2245.

[221] Schnitzspan, P., Roth, S., and Schiele, B. Automatic discovery of meaningful object parts with latent crfs. In *In CVPR* (2010), pp. 121–128.

[222] Schoelkopf, B., and Smola, A. J. *Learning with Kernels.* MIT Press, Cambridge, MA, 2002.

[223] Siddiqi, K., and Kimia, B. Toward a shock grammar for recognition. *CPVR:* (1996).

[224] Sivic, J., Russell, B. C., Efros, A. A., Zisserman, A., and Freeman, W. T. Discovering Objects and their Localization in Images. In *ICCV* (2005), pp. 370–377.

[225] Slaney, J., and Thiébaux, S. Blocks World revisted. *Artificial Intelligence*, 125 (2001), 119–153.

[226] Small, C. G. *The Statistical Theory of Shape.* Springer, New York, NY, 1996.

[227] Sneath, P. H. A. Trend-surface analysis of transformation grids. *Zoology 151*, 65–122.

[228] Srinivasan, P., Zhu, Q., and Shi, J. Manty-to-one Contour Matching for Describing and Discriminating Object Shape. *CVPR* (2010).

[229] Swain, M., and Ballard, D. Color Indexing. *IJCV 7*, 1 (1991), 11–32.

[230] Sýkora, D., Dingliana, J., and Collins, S. AS-rigid-as-possible image registration for hand-drawn cartoon. In *NPAR* (2009).

[231] Thompson, D. W. *On Growth and Form.* Dover, 1917.

[232] Todorovic, S., and Ahuja, N. Learning subcategory relevances for category recognition. In *CVPR* (2008).

[233] Toshev, A., Taskar, B., and Daniilidis, K. Object Detection via Boundary Structure Segmentation. In *CVPR* (2010).

[234] Treisman, A. Features and objects: The 14th Bartlett memorial lecture. *The Quarterly J. of Experimental Psychology* (1988).

[235] Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. Large Margin Methods for Structured and Interdependent Output Variables. *JMLR: 6* (2005), 1453–1484.

[236] Turk, G., and Levoy, M. Zippered Polygon Meshes form Range Images. In *In SIGGRAPH* (1994).

[237] Turk, M., and Pentland, A. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience 3*, 1 (1991), 71–86.

[238] Ulusoy, I., and Bishop, C. M. Generative versus discriminative methods for object recognition. In *CVPR:* (2005), pp. 258–265.

[239] Usami, Y., Stork, D. G., Fujiki, J., Hino, H., Akaho, S., and Murata, N. Improved methods for dewarping images in convex mirrors in fine art: Applications to van Eyck and Parmigianino. In *Computer vision and Image analysis of art II* (2011).

[240] Utcke, S. Error-bounds on curvature estimation. *Lecture Notes in Computer Science* (2003).

[241] v. Luxburg, U. Clustering stability: an overview. *Foundations and Trends in Machine Learning 2*, 3 (2010).

[242] Van de Sande, K., Gevers, T., and Snoek, C. Evaluating Color Descriptors for Object and Scene Recognition. *PAMI: 32*, 9 (2010), 1582–1596.

[243] Van de Sande, K., Uijlings, J., Gevers, T., and Smeulders, A. Segmentation as Selective Search for Object Recognition. In *ICCV* (2011).

[244] van de Weijer, J., Gevers, T., and Bagdanov, A. Boosting color saliency in image feature detection. *PAMI: 28*, 1 (2006), 150–156.

[245] Vedaldi, A., Gulshan, V., Varma, M., and Zisserman, A. Multiple kernels for object detection. In *ICCV* (2009).

[246] Vidal, R., Ma, Y., and Sastry, S. Generalized principal component analysis (gpca). *PAMI: 27*, 12 (2005), 1945–1959.

[247] Vijayanarasimhan, S., and Grauman, K. Efficient region search for object detection. In *CVPR* (2011).

[248] Viola, P. A., and Jones, M. J. Rapid Object Detection using a Boosted Cascade of Simple Features. In *CVPR* (2001), pp. 511–518.

[249] Wahba, G. *Spline models for observational data.* Society for Industrial and Applied Mathematics., 1990.

[250] Wahba, G. *Spline Models for Observational Data.* Series in Applied Mathematics (SIAM), Vol. 59, Philadelphia, PA, 1990.

[251] Waltz, D. Generating Semantic Description from Drawings of Scenes with Shadows. Tech. rep., 1972.

[252] Wang, H. Y., Yang, Q., and Zha, H. Adaptive p-posterior mixture models kernels for multiple-instance learning. In *ICML:* (2008).

[253] Wang, J., and Adelson, E. Representing moving images with layers. *IEEE Trans. on IP 3*, 5 (1994).

[254] Wang, X., Han, T., and Yan, S. An hog-lbp human detector with partial occlusion handling. In *ICCV* (2009).

[255] Weber, M., Welling, M., and Perona, P. Towards Automatic Discovery of Object Categories. In *CVPR* (2000), pp. 2101–2108.

[256] Weber, M., Welling, M., and Perona, P. Unsupervised Learning of Models for Recognition. In *ECCV* (2000), pp. 18–32.

[257] Wei, Y., and Tao, L. Efficient Histogram-Based Sliding Window. In *CVPR* (2010).

[258] Wells, W. M. Statistical Approaches to Feature-Based Object Recognition. *IJCV: 22*, 1 (1997), 63–98.

[259] Witkin, A. Scale space filtering. In *10th Int. Joint Conference on A.I.* (1983).

[260] Wolfe, J. Guided Search 2.0: A revised model of visual search. *Psychonomic Bulletin and Review 1* (1994), 202–238.

[261] Wolfe, J. M. Visual Search. In *Attention*, H. Paschler, Ed., university college of london press ed., vol. 21. 1998.

[262] Wolfe, J. M., Yee, A., and Friedman-Hill, S. R. Curvature is a basic feature for visual search tasks. In *Perception* (1992).

[263] Worring, M., and Smeulders, A. W. M. Digital curvature estimation. In *CVGIP: Image Understanding* (1993).

[264] Yanv, J., and Pollefeys, M. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. *ECCV:* (2006).

[265] Yarlagadda, P., Monroy, A., Carque, B., and Ommer, B. Towards a Computer-based Understanding of Medieval Images. In *Scientific Computing and Cultural Heritage (SCCH)* (2009).

[266] Yarlagadda, P., Monroy, A., and Ommer, B. Voting by Grouping Dependent Parts. *ECCV:* (2010).

[267] Yarlagadda, P., and Ommer, B. From Meaningful Contours to Discriminative Object Shape. In *ECCV:* (2012).

[268] Yuille, A., Hallinan, P., and Cohen, D. Feature Extraction from Faces using Deformable Templates. *IJCV 8*, 2 (1992), 99–111.

[269] Yuille, A., and Kosowsky, J. Statistical Physics Algorithms that Converge. *Neural Computation*, 6 (1994), 341–356.

[270] Yuille, A., and Rangarajan, A. The concave-convex procedure (CCCP. In *Advances in Neural Information Processing Systems 14* (2002).

[271] Zass, R., and Shashua, A. Probabilistic graph and hypergraph matching. In *CVPR:* (2008).

[272] Zhou, Z. H., and Xu, J. M. On the Relation between multiple-instance learning and semi-supervised learning. In *ICML:* (2007).

[273] Zhu, L., Chen, Y., Yuille, A. L., and Freeman, W. Latent hierarchical structural learning for object detection. In *CVPR* (2010), pp. 1062–1069.