Graz University of Technology

Institute of Electrical Measurement and Measurement Signal Processing

Master Thesis

Analysis Of Distance Transforms By Means Of Comparing Contours With Images

Georg Teichtmeister, Bakk.techn.

Graz, Austria, May 2010

Thesis supervisor

Ao. Univ.-Prof. Dipl.-Ing. Dr.techn. Axel Pinz

To …
my parents Doris & Franz Teichtmeister.

TU Graz
Graz University of Technology
Senat

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Graz, am …………………………… ………………………………………………..
(Unterschrift)

Englische Fassung:

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

…………………………… ………………………………………………..
date (signature)

Abstract

Recent object categorization methods use shape information to categorize objects in real world scenes. The shape is represented by the contour of the objects. This work analyzes methods to match contours or contour fragments to images.

After a survey on methods of contour and shape matching is given, this work concentrates on methods based on the distance transform and methods which create a similarity image using kernels. Furthermore a possible application of the earth mover´s distance to the problem at hand is presented.

A methodology to evaluate the methods by matching contour fragments to images is presented. This methodology is designed to evaluate these methods on how well they will perform for object categorization methods, without having the need for an actual implementation of an object categorization algorithm.

The results show, that the best method combines the already known average distance value of the contour fragment enhanced by orientation planes and the newly introduced estimation of the variance of the distances.

Keywords: distance transform, contour fragment, boundary fragment, contour matching, object categorization

Acknowledgments

I want to thank all the people who helped me with completing this thesis. Many of them are with or have been with the Institute of Electrical Measurement and Measurement Processing at the Graz University of Technology.

Especially my two colleagues Kerstin Pötsch and Peter Holzer who helped me with their advice, proof reading and encouraging me when I needed it.

A special thanks goes to my thesis advisor Prof. Axel Pinz. His deep knowledge and great experience in the field of computer vision was a great source of information for me.

Finally I want to thank my parents. Their support during the years of my studies finally led to this thesis.

Contents

Chapter 1

Introduction

In this chapter the problem of comparing contours with images is first motivated by its application in object categorization. Then the problem and the goals of this work are defined and finally the major part of this chapter covers related work on the topic.

1.1 Motivation

In recent years two contributions, Opelt et al. [20] and Shotten et al. [26], made use of the external shape of objects to categorize them. As the extraction of a continuous and closed shape is difficult (within clutter) to impossible (in case of occlusions), both contributions used fragments of the object boundary [20] or contour [26], resulting in models of multiple boundary or contour fragments.

A central part of both contributions is the matching of contours to images. Both used the chamfer matching method, presented by Borgefors [3], to solve that problem. Opelt et al. [20] added information about edge orientation using orientation planes, Shotton et al. [26] introduced an argument distance transform to use edge orientation as additional matching metric. While adding information about edge orientation to chamfer matching was an important step, both contributions focused on object categorization, and tried to improve object categorization by using more sophisticated learning algorithms and object representations.

The aim of this work is an analysis and improvement of one of the foundations of above mentioned contributions: The matching of contour fragments to images.

The application of chamfer matching on real world images, as Opelt et al. [20] and Shotton et al. [26] did in their contributions, yields poor results in terms of localization and also w.r.t. the question if there is an object of a particular category at a particular location. In figure 1.1 an example is shown: Only a slight distortion results in a completely wrong location for the best hit. The correct hit (red in the center, rear legs of the cow) would be ranked fifth. To compensate this drawbacks, both contributions [20] and [26] used advanced learning algorithms. If contour matching could be improved, better results should be achieved in object categorization.

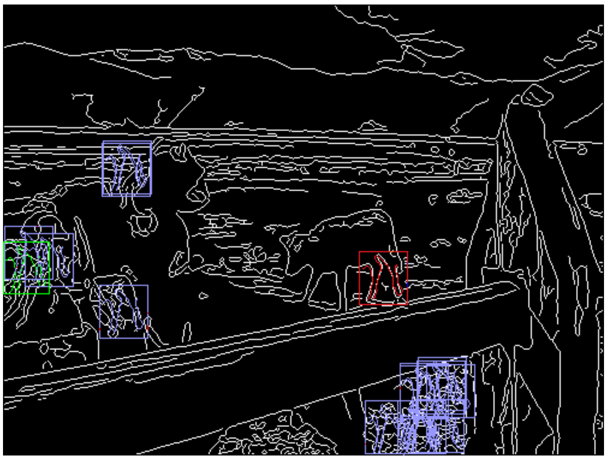

Figure 1.1: 20 best hits in a cow image. The green hit (left side) is the best hit, the red one (center) is the ground truth location. The contour fragment is rotated and skewed by an angle of -5°. The original image is seen in figure 3.3

.

It is also seen, that in the bottom center of figure 1.1 many hits are located within clutter, which stems from texture on the ground. Chamfer matching performs bad in clutter, because the distance to the next edge pixel is low, because there are so many edge pixels. However, using simple edge detection algorithms, clutter is inevitable (e.g. leaves, grass, highly textured clothes). Therefore the robustness to clutter is a key property of contour matching methods.

The second goal of the thesis is to analyse the behaviour of methods, which match contours to images. Knowing weaknesses of the used methods, makes it easier to compensate them with higher level methods. One characteristic of object categorization is the intra-class variance. This means, that objects of the same category can actually look quite different. In the case of using the shape of an object as descriptor, it is beneficial to know, to which point the contour matching method can capture shape variance, and at which point of shape variance the object categorization method has to take care of it.

1.2 Definition of the problem

This work wants to give answers to two questions: First, how is a given contour fragment matched to an image, and second, how can different methods, solving the first problem, be compared to each other.

The basic problem is finding a given contour fragment in an image. The contour fragment is given as a binary image, the image is a greyvalue image and the result should be some sort of similarity or distance function (the pair similarity/distance will be called quality from now on). This quality function should tell the quality of match between the contour fragment and the image on each image location. From the quality function just the best match, the n best matches or all extrema which fulfill a certain criterion can be chosen as hits of the contour fragment on the image.

Due to the intra class variance of object categorization, corresponding contours on objects of the same class are just similar not the same. Therefore, a key requirement is the ability of the matching metric to reflect that fact. For example the naive approach of correlating a contour with an edge image would yield a perfect score for a good match, but when the match becomes non-perfect the score drops rapidly.

Other preferable properties of the resulting quality function are smoothness and a low number of extrema. If a non-exhaustive search is used, these properties would improve the time needed to find extrema, and even if an exhaustive search is used, the number of extrema that have to be checked is reduced, which also leads to a faster runtime.

Beside the intra class variance of object categories, contours are also translated, rotated and scaled, because of different imaging conditions. Invariance to translation is achieved by scanning the whole image with a translated contour, resulting in an exact location. Invariance to large differences in scale and rotation also require a search with a scaled and/or rotated version of the contour. But methods for matching contours to images should be robust to small changes in rotation and scale, because exact rotation and scale are not useful to categorize and localize an object.

To compare different methods for the task of matching contours to images, a measurement has to be found, from which the quality of a certain method can be derived. Such a measurement should be easily computed from the quality of match between a contour and an image and easily reproduced by other authors. Furthermore, it should also give some insight into the properties of the measured methods.

1.3 Related work

To the end of 1990ies most shape matching was done on already segmented shapes. One example is the review of Veltkamp and Hagedoorn [29], another one Zhang and Lu [31], which gives a good overview on shape representation.

Both contributions present shape matching methods which require an already segmented object. There are three groups of methods: Methods which match regions by shape (e.g. [18, 22, 23, 28]), methods which match closed contours (e.g. [7, 19, 30]) and methods which can be applied to shape fragments (e.g. [2]).

The requirement of a prior segmentation step eliminates the possibility to use these methods for an improvement of object categorization algorithms like [20] or [26], as the segmentation of the object is part of their task.

In the remainder of this section, a more detailed review is given on the methods which are compared later in the thesis, as well as on the Hausdorff distance and on multilayered shape representations. The Hausdorff distance was already compared to chamfer matching in [3] and [10] and found to be inferior to the chamfer distance.

Multilayer shape representations are a different and promising approach to matching contours to images. A comparison between distance transforms and multilayer shape representations within one experimental framework could probably not exploit the advantages of both groups of methods. Therefore, multilayer shape representations are left out of the comparison and only mentioned for completeness.

1.3.1 Chamfer Matching

As in the 1970´s the problem of matching maps with aerial pictures arose, chamfer matching was invented by Barrow [1], to solve that problem. It was improved by Borgefors in [3] and [4]. Basically, the chamfer distance between a contour and the edge map of an image, is the average minimum distance between the contour pixels and the pixels of the edge map.

The chamfer distance is efficiently computed by computing the distance transform of the edge map and then summing up the values of the distance transform along the shifted contour.

The distance transform of a binary image computes the minimum distance of each pixel to the “one” - pixels. The underlying distance can be freely chosen, for chamfer matching the so called chamfer distance is used, which is a good approximation of the Euclidian distance on rasterized images. It is easily seen, that the distance transform at the locations of the “one” - pixels is zero.

Both, Shotton et al. [26] and Opelt et al. [20], use a modified version of chamfer matching in their contributions to object categorization. They improved it by adding orientation information to the distance. Opelt used an idea originally suggested by Gavrila [15] to split the original edge image up into orientation channels, where one channel contains only the edges within a certain orientation range. When computing the chamfer distance, only the corresponding orientation channels are matched. The final result is then the sum over all channels.

On the other hand, Shotton et al. introduced the argument distance transform, where each pixel is assigned the orientation value of the nearest edge pixel. The orientation distance is the average difference between the orientations of the contour pixels and the pixels in the edge map. They finally build a weighted sum between the orientation distance and the chamfer distance.

1.3.2 Edge Potential Functions

Dao et al. [9] introduced the notion of an edge potential function, where each edge pixel is analog to an electrical charge, and the edge potential function is computed like the electrical potential function. At the matching step, a template is correlated with the edge potential function and maxima correspond to good matching locations.

As with the chamfer distance, the edge potential field is a smooth function, however it is a similarity function in contrast to the distance transform of chamfer matching. A drawback of similarity functions is the lack of an absolute optimum, in contrast to distance functions, where the absolute optimum is zero.

1.3.3 Hausdorff Distance

Huttenlocher et al. [17] introduced the Hausdorff distance into the field of shape matching. The Hausdorff distance is best explained in two steps: First the directed Hausdorff distance is computed between two contours, from contour A to B and the other way round. In the second step the larger of the two distances is chosen to be the Hausdorff distance. The directed Hausdorff distance first computes the minimum distance to the points of B for each point of A. The maximum of these distances is chosen as the directed Hausdorff distance.

Huttenlocher et al. also suggested an efficient algorithm to compute the directed Hausdorff Distance when searching a template in an image under an unknown translation:

Basically one computes the Voronoi surface (with respect to the underlying norm) of the image and then translates the template over the Voronoi surface and then chooses the maximum (or p-Quantile) from the points corresponding to the template edge points.

Knowing that a distance transform is a Voronoi surface, and using the mean instead of the maximum one can see the relations between chamfer matching and the directed Hausdorff distance quite easily. Borgefors [3] actually suggested a chamfer distance using the maximum instead of the average. Other authors [10, 32] came up with similar ideas but calling it mean Hausdorff distance, or modified Hausdorff distance. Borgefors [3] and Dubuisson and Jain [10] compared several methods and both contributions reported that the partial Hausdorff distance (using the maximum value) is inferior to the chamfer distance (or partial Hausdorff distance using the mean value). Borgefors also reports, that the same is true for the median.

1.3.4 Earth Mover´s Distance

The Earth Mover´s Distance (EMD) was introduced by Rubner et al. [25] to compare color histograms. The basic idea is to measure the distance a unit of dirt has to be transported from a pile to a hole. Interpreting a color histogram as a collection of dirt piles and another one as a collection of holes, the EMD can be used to compare these two histograms.

Grauman and Darrell [16] introduced the EMD to the field of shape matching. Even if they match presegmented shapes, the idea of the EMD is worth to be examined within the scope of this thesis.

1.3.5 Multilayer Shape Representation

Inspired by biological vision, multilayered shape representations are suggested for use in object categorization. Riesenhuber and Poggio [24] analyze neural mechanisms in the visual cortex and connect them to computer vision.

A purely computer vision based approach is suggested by Fidler et al. [12]. On the basis of oriented Gabor filters (an example is given in figure 1.2), which detect short lines (parts), longer contours are represented as combinations of these parts. Fidler et al. introduced a layered hierarchy, where a part in each layer is a combination of three parts of the underlying layer. Layer 1 are the responses of the Gabor filters. For each combination of three parts, the three parts, their relative locations and orientations, and the variance of the relative locations are stored. Also, learning the variance of the relative locations gives a more flexibel model, so that small variations do not lead to different parts, but are seen as different instances of the same part.

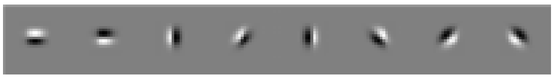

Figure 1.2: Example of layer 1 Gabor filters as used in [12]

In subsequent works [13, 14] Fidler et al. extended the idea of hierarchical representations to a framework for object categorization.

1.3.6 Summary

Chamfer Matching and Eletrical Potential Functions are the most promising approaches for the problem at hand. Both methods solve the problem directly and inherently, as they take a contour fragment and an edge image as input. In contrast to these two methods are shape matching methods, which take two shapes as input and compute a distance between two shapes. To get shapes from an image, it has either to be segmented beforehand or shapes have to be extracted from the edge image. Both approaches are not reasonable, because segmenting the image before the matching would solve the object categorization problem, and extracting shapes from an edge image is computationally infeasible. The set of shapes to be extracted from a an edge image is a power set of the set of edgels, with cardinality 2^N, N being the number of edgels in the edge image.

The application of the Earth Mover´s Distance to matching contours to images, as presented in chapter 2, allows the direct compution of the EMD between a contour and an edge image. Therefore, it is also suited to be applied to the problem at hand.

For these reasons, chamfer matching, electrical potential functions and EMD are reviewed in the remainder of this thesis. Matching contours to images based on the Hausdorff distance has already proven to be inferior to chamfer matching [3] and [10] and multilayer shape representations, which learn contours by combining small pieces of edges, would not utilize all their advantages in an experimental framework targeted at methods using a given contour fragment.

Chapter 2

Methods

This chapter illustrates the different methods employed in matching a contour to an image. The inputs to the process of matching are a greyscale image and a binary representation of the contour image. The image is preprocessed to either a gradient image or a binary edge image. After that the contour fragment is matched to the image. For some methods this matching process can be split into a contour fragment independent part and a much cheaper (w.r.t. computational complexity) part, which is dependent on the contour fragment. These methods are the distance transform methods and the methods employing a kernel to get similarity images. After computing these distance or similarity images, the contour fragment is matched to them, computing the mean of the distance or similarity image along the contour. It is also examined, if the computation of higher order statistical moments would help in the matching of contour fragments to an image.

For the Earth movers distance (EMD), this splitting is not possible, so there are no means to alter the computation. However, this is not desirable, because the basic idea of the EMD is different to those which the distance transform and kernel based methods rely on.

Finally, the above mentioned methods are extended to analyze the influence of information about the orientation of edges. It has already been seen [20, 26] that orientation information improves the detection rate of contour fragments.

2.1 Distance Transform

A distance transform $dt(I)$ of a binary image I gives a two dimensional function, which gives the minimal distance to all the "ones" in the binary image. Examples are given in figures 2.1 and 2.2.

Using a distance transform of an image, the distance to the nearest “one” pixel at a given location can be determined in $O(1)$. In the case of matching a contour fragment cf to an image, the estimation of the average distance of its pixels to the respective nearest pixels of the binary image takes only $O(length(cf))$ time. When matching (see section 2.3 for details) many contour fragments to one image this is cleary advantageous.

The remainder of this section gives a mathematical formulation of the distance transform as well as possible extensions and modifications to it, and finally discusses implementation details.

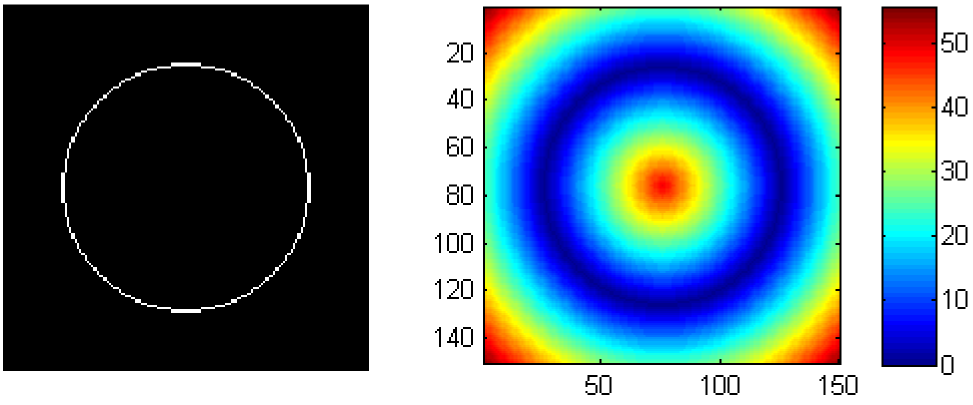

Figure 2.1: A circle (to be interpreted as an edge image) on the left side and its distance transform to the right.

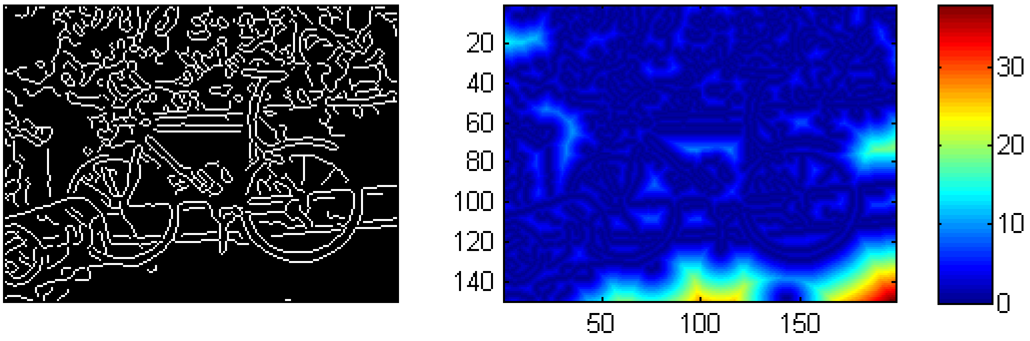

Figure 2.2: This is the edge image (left) and its distance transform (right) of a bike image.

2.1.1 Definition

The distance transform $dt_{\boldsymbol{x}}(I)$ at location $\boldsymbol{x}$ of an image I is given as

$$dt_{\boldsymbol{x}}(I) = \min_{I(\boldsymbol{y})=1} |\boldsymbol{x} - \boldsymbol{y}|, \tag{2.1}$$

where $\boldsymbol{y}$ are the locations of the "one" pixels in the image I and $|\cdot|$ depicts an arbitrary norm. In most applications of the distance transform this norm is the Euclidian distance.

2.1.2 Modifications

Actually there are two possibilities to modify the distance transform as presented above: On the one hand, the norm used for the distance computation could be changed, on the other hand, a nonlinear function could be applied to the result of the Euclidian distance transform.

Modifying the norm when computing the distance transform leads to effects which are not desirable for most applications. An L_p - norm different to the Euclidian L_2 - distance introduces a dependence on the orientation, which results in different distance transforms for different rotations of the same edge image (see equation 2.2). This is easily seen if one looks at the unit distance under different norms in figure 2.3 and the example with a line in figure 2.4.

$$dt(I) \neq rot(dt(rot(I, \varphi)), -\varphi) \tag{2.2}$$

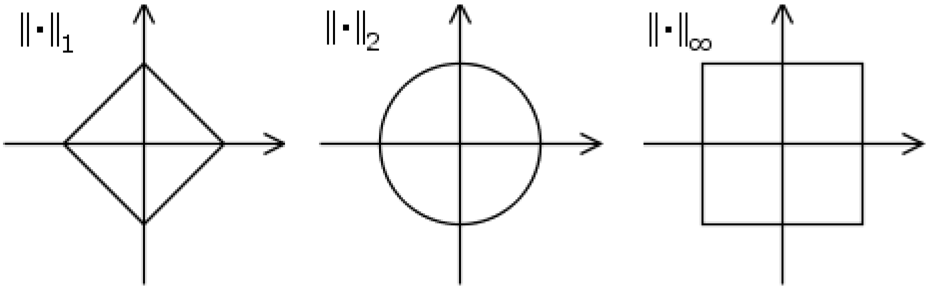

Figure 2.3: The unit circle under the L_1, L_2 and L_∞ norm.

The second option is to apply a nonlinear function to the Euclidian distance transform. Obviously these functions should be continuous and monotonically increasing. With these properties they only stretch or compress the original Euclidian distance. However,

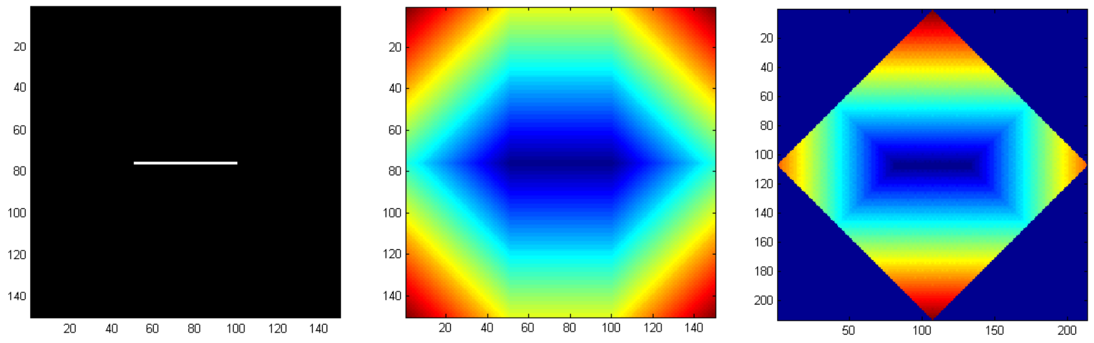

Figure 2.4: The center image shows the distance transform using the cityblock distance (L_1 - norm) of the line in the left image. To generate the right image, the line is rotated by $\frac{\pi}{4}$, the distance transform is computed and the image containing the distance transform is rotated back so it can be compared to the center image.

combined with the linear matching step they can lead to quite different results.

The easiest nonlinear function is obvioulsy a polynomial $f(x) = (\frac{x}{a})^p$ for arbitrary p. Parameter a controls the point where the polynomial is equal to the linear map. Figure 2.5 shows different polynomials. It is easy to see, that a polynomial condenses distances between values $x < a$ and stretches them for $x > a$.

2.1.3 Methods and Implementation

It is obvious, that computing the distance to every "one" pixel and then choosing the minimum for each location is not the best solution in terms of computational efficiency. Barrow et al. [1] used a two-pass method for computing the distance transform, called chamfering. Because of this, Barrow et al. [1] and Borgefors [3] called their contour matching methods chamfer matching. The basic idea of the chamfering method is the forwarding of local information to neighbouring pixels. Therefore, this method is linear w.r.t. the size of the image.

Breu et al. [5] presented a method, also linear w.r.t. the size of the image, which relies on Voronoi diagrams to compute the distance transform (or nearest-neighbour transform as Breu et al. call it). This algorithm is actually used to conduct the experiments for the thesis at hand.

A detailed comparison of methods to compute the distance transform was published by Paglieroni [21].

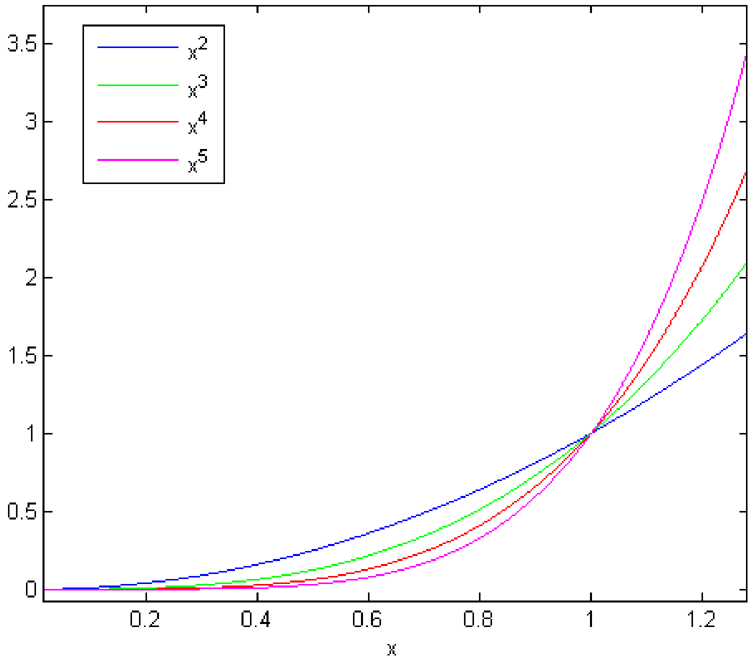

Figure 2.5: Polynomials x^p for different p

2.2 Kernel based similarity methods

Dao et al. [9] introduced the idea of edge potential functions. In this section, their basic idea is explained. Subsequently the transformation of their idea into a mathematical framework more common in the field of computer vision is described. Finally, different kernels are presented.

2.2.1 Edge Potential Fields

The basic idea of edge potential fields (EPFs), as presented in [9], is, that each edge pixel represents an (electrical) charge and a potential field is constructed as the sum of the effects of all charges, similar to an electrical potential field. The electrical potential field ϕ of discrete charges Q_i in vacuum is given as

$$\phi(\boldsymbol{r}) = \frac{1}{4\pi\epsilon_0} \sum_i \frac{Q_i}{\parallel \boldsymbol{w_i} - \boldsymbol{r} \parallel}, \tag{2.3}$$

where $\boldsymbol{r}$ is the observation point, ϵ_0 the electrical permittivity and $\boldsymbol{w_i}$ are the locations of the charges, and $\| \cdot \|$ denotes the Euclidian distance.

Similarly, the EPF at location $\boldsymbol{x}$ is given as

$$EPF(\boldsymbol{x}) = \frac{1}{4\pi\epsilon_{eq}} \sum_i \frac{Q(\boldsymbol{y_i})}{\| \boldsymbol{y_i} - \boldsymbol{x} \|}, \tag{2.4}$$

where $Q(\boldsymbol{y_i})$ is the charge at location $\boldsymbol{y_i}$ and ϵ_{eq} is an equivalent to the eletrical permittivity. This parameter is of some importance, because it controls the smoothness of the potential field. A smaller ϵ_{eq} results in a smoother potential field.

One obvious problem of the above equations is the singularity at the location of the charge. In physics this is obviously not a practical problem, because at small distances the laws of classic physics get less important to the favour of the laws of quantum physics. However, in the application of the electrical potential field to edge images, there is a need to compute an edge potential value for the edge location and this value should be reasonable for subsequent processing steps, like edge matching. Dao et al. suggested to clip the EPF at those locations, but did not give any actual values.

The last detail is the choice of a value for the charge Q in an edge image. On binary images a constant value (like one) is the obvious choice. However, Dao et al. [9] also suggested to use the magnitude of the gradient at the edge location as value for the charge Q using additional information about the strength of an edge.

2.2.2 EPF to Kernel

Either from the analogy to physics or mathematically, it is obvious that one cannot only look at a specific location and add up all effects of the charges, but also look at all the charges, compute their effects and add them up.

$$EPF_s(\boldsymbol{x}) = \frac{1}{4\pi\epsilon_{eq} \| \boldsymbol{x} \|} \tag{2.5}$$

$$EPF(\boldsymbol{x}) = \sum_i Q_i \cdot EPF_s(\boldsymbol{x} - \boldsymbol{y_i}) \tag{2.6}$$

$$EPF(\boldsymbol{x}) = \sum_{\boldsymbol{i}} I(\boldsymbol{y_i}) \cdot EPF_s(\boldsymbol{x} - \boldsymbol{y_i}) \tag{2.7}$$

Starting from equation 2.4, one can express the effect of one charge with equation 2.5 and sum up all effects in equation 2.6. Rewriting equation 2.6 leads to equation 2.7, which expresses the computation of the EPF as a convolution of the image with a single kernel.

Viewing the EPF in this way not only helps in implementing it, but it is also a key step to understand the algorithm of Dao et al.: An EPF is the convolution of the edge image with a kernel constructed of a $\frac{1}{r^2}$ function, $r = \parallel \boldsymbol{x} \parallel$ being the distance to the charge. As a last detail this kernel represents a low pass filter.

Now having this idea of convolving the edge image with a low pass filter, one could think of other filter kernels than this $\frac{1}{r^2}$ filter.

2.2.3 Possible realizations of kernels

Possible realizations of a kernel are the standard kernel described in equation 2.5, the well known Gaussian kernel, as well as kernels using a linear and constant fall off function.

The Gaussian kernel, described in equation 2.8 is parametrized by the standard deviation σ. By using only a scalar as parameter, the Gaussian kernel is circular in shape as can be seen in figure 2.6.

$$g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.8}$$

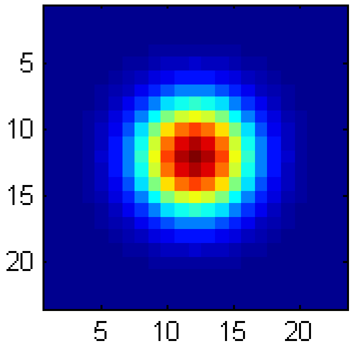

Figure 2.6: Example of a Gaussian kernel.

2.3 Matching of contour fragments to distance and similarity images

The result of a distance transform or the convolution of the edge image with a kernel, is either a distance image or a similarity image. To simplify the further discussion, those two terms are merged into the term "quality image". As already said, a quality image is the result of splitting the matching process into a contour fragment independent part, and one which is dependent on the contour fragment.

When matching a contour fragment to a quality image, a matching score is computed for each location in the image. A better score corresponds to a better match. One of the more obvious choices to compute a score is to compute the average quality along the contour. A simple method to achieve this, is the cross correlation between the the contour fragment c and the quality image q averaged by the length N of the contour. The score of the match m is given as

$$m(x, y) = \frac{1}{N} \sum_{u=1}^{U} \sum_{v=1}^{V} q(x + u, y + v) c(u, v), \tag{2.9}$$

where U, V are the size of the bounding rectangle of the contour. In this case the cross correlation is equivalent to summing up all quality values “below” the contour, because the contour is a binary image. This is also the notation of Barrow et al. [1] and Borgefors [3, 4]. The above - in the context of Barrow and Borgefors - means, that the average distance between the contour pixels and the “one” pixels in the edge image should be minimal.

Another view would be a statistical one. Seeing all the quality values “below” the contour fragement as a distribution of qualities, the mean quality could be computed. Now there is not only the mean value, but many more features which could be used to classify a distribution. Borgefors already discussed the mean, median, root mean square and maximum in [3].

Continuing this line of thought, the variance could be an interesting feature, at least supporting the mean, as it could be believed that matches with small variance in quality are better than ones with big variance.

2.4 Earth movers distance

The Earth movers distance (EMD) was introduced by Rubner et al. [25]. Their initial idea was to suggest a better way to match histograms. In their paper two ideas are presented: The first idea was not only to match corresponding bins of a histogram, but to try to compare the histograms as a whole, which was the EMD. The second idea was not to compute a histogram with fixed sized bins, but to cluster the data to so called signatures, which optimized the trade-off between efficiency and expressiveness. Converting the edge image and the contour fragment to signatures, it is easy to compute the EMD between the contour fragment and the edge image.

2.4.1 Signatures

Rubner et al. [25] introduced signatures to achieve two things: A better expression of the data, and a more memory efficient representation. The former means that the loss of information is minimized, and the latter a reduction in memory space. If one works with greyvalue images, these two concerns may not be relevant, because it is relatively cheap w.r.t. memory to store a perfect representation of a histogram. If someone works with color histograms, as Rubner et al. did, saving a perfect representation of an image would be more costly in terms of memory. Even if in a color image not all of the 16.7 million colors will be used, the number of colors exceeds 100,000 most of the time.

To overcome these problems, Rubner et al. suggested to cluster the data and to generate signatures which are represented as sets of pairs $s = (\boldsymbol{x}, w)$, where w is the weight at location $\boldsymbol{x}$. By that, a complete signature S is given as $S = \{s_j = (\boldsymbol{x_j}, w_j)\}, j = 1 \dots N$, where N is the number of pairs.

2.4.2 Computing the EMD between signatures

To compute the distance between two signatures, Rubner [25] called one signature "dirt piles" and the other one "holes". The EMD is the minimum work needed to move the dirt from the piles into the holes. Work is just the product of dirt by ground distance. Where ground distance is the distance between the location of the pile $\boldsymbol{x_p}$ and the location of the hole $\boldsymbol{x_h}$ ($\| \boldsymbol{x_p} - \boldsymbol{x_h} \|$).

Before the minimum work - or earth movers distance - can be computed, the best way to move the dirt piles into the holes has to be found. This is called the transportation problem. It describes the problem for transporting goods from suppliers (piles) to consumers (holes).

To better understand the EMD formulation, a short overview to the transportation problem (see also figure 2.7) as given in [25] is presented: Let $\mathcal{I}$ be a set of suppliers and $\mathcal{J}$ a set of consumers, and c_{ij} the cost of transporting one unit of goods from supplier $i \in \mathcal{I}$ to consumer $j \in \mathcal{J}$, then the goal is to find a flow f_{ij} which minimizes the overall transportation cost C

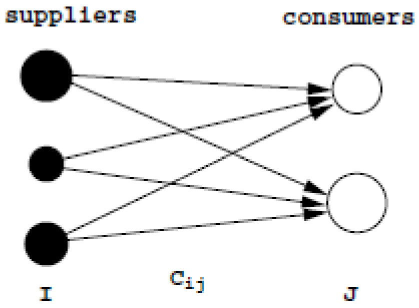

Figure 2.7: An example of a transportation problem from [25].

$$C = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} c_{ij} f_{ij}, \tag{2.10}$$

subject to following constraints,

$$f_{ij} \geq 0 \qquad i \in \mathcal{I}, j \in \mathcal{J} \tag{2.11}$$

$$\sum_{i \in \mathcal{I}} f_{ij} = y_j \qquad j \in \mathcal{J} \tag{2.12}$$

$$\sum_{j \in \mathcal{J}} f_{ij} \leq x_i \qquad i \in \mathcal{I}, \tag{2.13}$$

where y_j is the total capacity of consumer j, and x_i is the total supply of supplier i. Goods should only be transported from suppliers to consumers (constraint 2.11), each consumer's needs should be satisfied (constraint 2.12), and the supply of a supplier must not be exceeded (constraint 2.13). As consequence of these constraints the total supply must exceed the total consumption

$$\sum_{j \in \mathcal{J}} y_j \leq \sum_{i \in \mathcal{I}} x_i. \tag{2.14}$$

Applying the transportation problem to signatures is quite straightforward: One signature is the supplier and the other one is the consumer. To satisfy condition 2.14, the signature with the smaller total weights is the consumer. By solving the transportation problem, the optimal flow is found. Then the earth movers's distance is given by

$$EMD(\mathcal{I}, \mathcal{J}) = \frac{\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} c_{ij} f_{ij}}{\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} f_{ij}} = \frac{\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} c_{ij} f_{ij}}{\sum_{j \in \mathcal{J}} y_j} \tag{2.15}$$

The denominator is a normalization factor to even out the influence of the signature total weight. Signatures with smaller total weight would be favoured (having a smaller distance) otherwise.

2.4.3 Applying EMD to edge images

The basic idea to apply the EMD to edge images is quite simple: The edge image and the contour are both converted to signatures and the EMD is computed. To convert an edge image (or the edge image representation of the contour fragment) to a signature, each pixel p will be represented by its location $\boldsymbol{x}$ and its weight w. The signature of an edge image E with N edge pixels is then defined as

$$E = \{p_i = (\boldsymbol{x_i}, w_i)\} \qquad i = 1 \ldots N. \tag{2.16}$$

The weight is either $w = 1$ if the edge information is used only, or the actual magnitude of the gradient at that edge pixel.

Finally, two details have to be mentioned: When shifting the contour over the image, the signature corresponding to the contour has to be computed at every step. It is obvious that this can be done very efficiently, because the signature locations only have to be updated with the relative movement of the patch. The second detail is a performance issue: Solving the transportation problem with the simplex algorithm [8] needs polynomial time in the average case w.r.t. the number of signatures. Matching the contour to the whole image, which has around 5,000 to 10,000 edge pixels, is computationally infeasible. Therefore, the shifted contour patch is only matched to the corresponding portion of the image.

2.5 Edge orientation

As already pointed out by Shotton et al. [26] and Opelt et al. [20], the edge orientation is a good way to improve the matching of contours to images. The basic idea is to compare the orientations of the contour and the edge in the image, to obtain a fitter cue if this would be a good match. The methods of Shotton et al. and Opelt et al. and a way to include edge orientation information into kernels for similarity images is described. To illustrate the different methods, an edge image containing a circle is used as an example. In figure 2.8 the circle and the orientation of the pixels are shown.

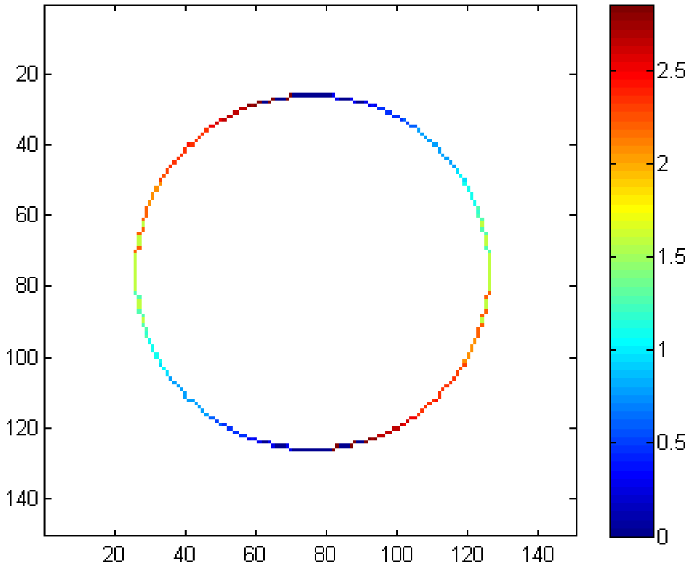

Figure 2.8: Orientation of edgels on a circle, given in radians.

2.5.1 Oriented Chamfer Matching

Shotton et al. [26] suggested a method called oriented chamfer matching. They augment the original chamfer matching term with an additive orientation matching term. In their work, the matching score $m(x, y)$ from equation 2.9 is being augmented by the orientation score $m_o(x, y)$. The combined score $m_c(x, y)$ is then

$$m_c(x, y) = m(x, y) + \lambda m_o(x, y), \tag{2.17}$$

where λ is used to control the sensitivity of the combined score to the orientation score. To compute the orientation score, Shotton et al. [26] just compare the orientation of the contour fragment edge pixel with that of the nearest image edge pixel. An efficient way to do this, is the argument distance transform $ADT(\boldsymbol{x})$. It works similar to the standard distance transform

$$ADT(\boldsymbol{x}) = \arg\min_{\boldsymbol{p}\in E} \| \boldsymbol{x} - \boldsymbol{p} \|_2, \tag{2.18}$$

where E is the set of edge pixels in the image. An example of the ADT is seen in figure 2.9. Using the ADT, the orientation score m_o between the the edge image E and the contour C can be computed

$$m_o(\boldsymbol{x}) = \frac{1}{N} \sum_{p\in C} |o(p) - ADT_E(\boldsymbol{p} + \boldsymbol{x})|, \tag{2.19}$$

where N is the length of the contour and $o(p)$ the orientation of edgel p.

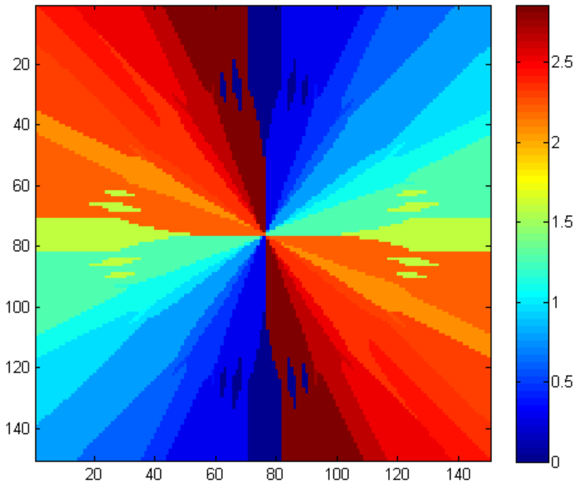

Figure 2.9: Argument distance transform of a circle, angle given in radians.

2.5.2 Orientation planes

Opelt et al. [20] used a method originally suggested by Gavrila [15]. The idea is to split up a contour or edge image into several images (called orientation planes), where each plane only contains the pixels of a certain range of orientation. In figure 2.10 an example is given, in which a circular contour is split up into orientation planes using Opelts parameters. Opelt et al. used eight orientation planes with an overlap of five degrees. Therefore, each orientation plane covers $180/8 + 2 * 5 = 22.5$ degrees. The boundaries for orientation plane i are

$$\begin{aligned} b_l &= 22.5i - 5 \\ b_u &= 22.5(i+1) + 5, \end{aligned} \tag{2.20}$$

where i starts at zero, b_l is the lower boundary and b_u the upper boundary.

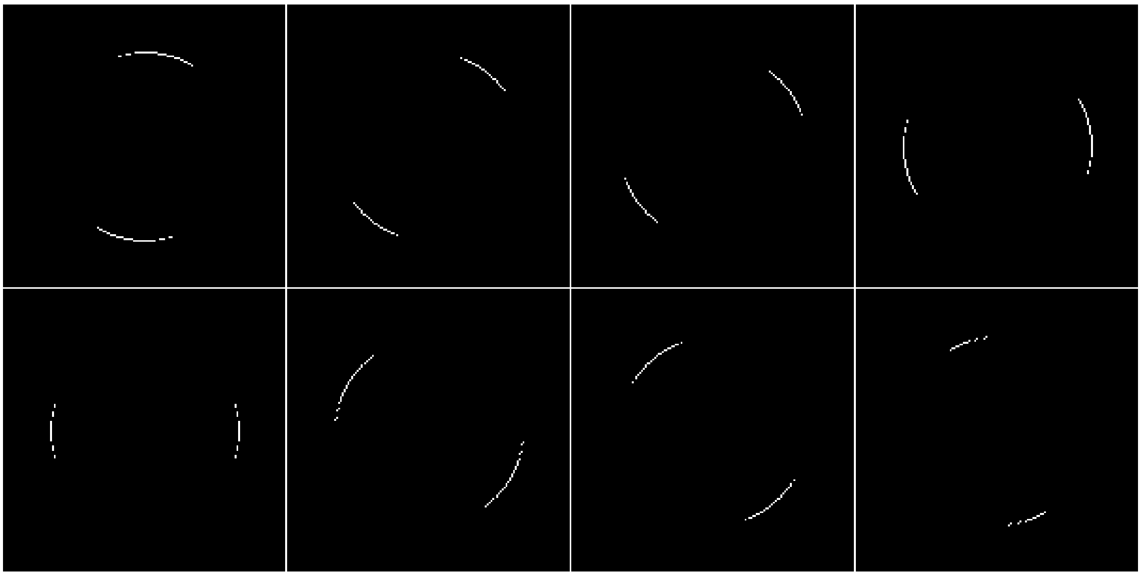

Figure 2.10: A circle decomposed into eight orientation planes with five degrees overlap.

To match the contour to the edge image, only the corresponding orientation planes are matched. To obtain the final result, the matching results for each orientation plane are added up.

2.5.3 Oriented kernels

Obviously, both of the above mentioned methods can also be used in combination with kernel based similarity images. But with kernels, the orientation information can be incorporated directly into the kernel and does not need to be applied later on. The key idea is to generate a separate oriented kernel for each edge pixel, instead of using a uniform kernel. In section 2.2.3 one Gaussian kernel was introduced (see equation 2.8). Although two dimensional, it has only one scalar σ which results in a circular, isotropic Gaussian kernel. Using a full covariance matrix allows to create all possible Gaussian distributions. The starting point to compute the covariance matrix are two vectors, one points into the direction of the edge, the other one perpendicular to it. The covariance matrix is easy to construct, if these two vectors are interpreted as its eigenvectors. Starting with the basic definition of an eigenvector $\boldsymbol{e}$ and its eigenvalue λ of matrix A

$$A\boldsymbol{e} = \lambda\boldsymbol{e}, \tag{2.21}$$

and knowing the two eigenvectors $\boldsymbol{e^{(1)}}, \boldsymbol{e^{(2)}}$ and setting their corresponding eigenvalues λ_1, λ_2, a system of linear equations can be constructed,

$$\begin{pmatrix} e_1^{(1)} & e_2^{(1)} & 0 & 0 \\ 0 & 0 & e_1^{(1)} & e_2^{(1)} \\ e_1^{(2)} & e_2^{(2)} & 0 & 0 \\ 0 & 0 & e_1^{(2)} & e_2^{(2)} \end{pmatrix} \begin{pmatrix} c_{11} \\ c_{21} \\ c_{12} \\ c_{22} \end{pmatrix} = \begin{pmatrix} \lambda_1 e_1^{(1)} \\ \lambda_1 e_2^{(1)} \\ \lambda_2 e_1^{(2)} \\ \lambda_2 e_2^{(2)} \end{pmatrix}, \tag{2.22}$$

where c_{ij} are the elements of the covariance matrix. Solving this system of linear equations is straight forward and the covariance matrix C is found. Using this covariance matrix, a Gaussian kernel can be constructed

$$g(\boldsymbol{x}) = \frac{1}{2\pi|C|^{\frac{1}{2}}} e^{-\frac{1}{2}\boldsymbol{x}^T C^{-1}\boldsymbol{x}}, \tag{2.23}$$

where $|\cdot|$ denotes the determinant of a matrix. Equation 2.23 constructs a Gaussian kernel which is centered at $\boldsymbol{x} = \boldsymbol{0}$ and is only aimed at two dimensional kernels. An example of such a kernel is given in figure 2.11.

2.6 Searching for good hits

Finally, one wants to know at which location the contour fragment matches best to the image contours. At first, local extrema are searched: Either minima for distance images, or maxima for similarity images. At these locations, the contour fragment has the best local match w.r.t. a certain matching score as described above. The second step would be the selection of either the best n local extrema, or local extrema satisfying some threshold criterion.

To find the local extrema, the Matlab implementation of the algorithm described in [27] is used.

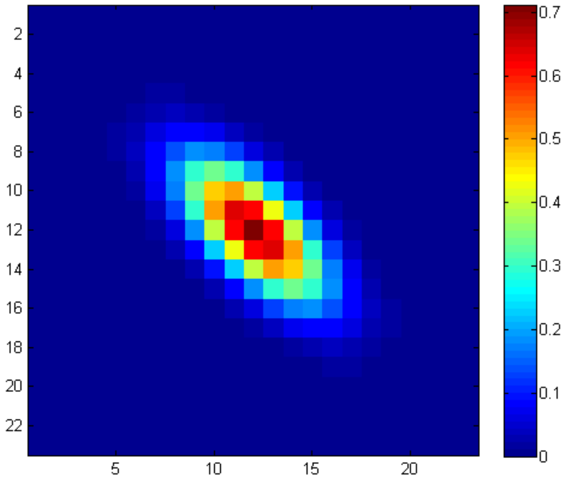

Figure 2.11: Example of an oriented Gaussian kernel with $\frac{\lambda_1}{\lambda_2} = 5$ and an angle of $\frac{\pi}{4}$.

Chapter 3

Experimental Setup

In chapter 2, methods for matching a contour fragment to an image were presented. The thesis at hand seeks to answer the question, which of these methods is the best for object categorization. This chapter discusses how to answer such a question.

The choice of the experimental data is discussed in the first section of this chapter. Choosing the right images is crucial to prove the point of the work. There are no artificial images, only images used in proven databases. The second part of the experimental data are the contour fragments. For the purpose of a perfect match, the contour fragments are extracted from the chosen images and then transformed to simulate different viewing geometries and intraclass variance.

The detailed explanation of the experiments is discussed in section 3.2. This section explains how the methods from chapter 2 are combined to produce actual results.

Finally the results of the experiments have to be evaluated and measurements of quality are presented in section 3.3. These are used to compare the different methods and give an answer to the question which method (or combination of methods) best matches a contour fragment to an image.

3.1 Experimental data

This section discusses the choice of the experimental data. The experimental data for the thesis at hand consist of the contour fragments and the images to match them. The first part presents the images which are chosen. The second part discusses the generation of the contour fragments to be matched to the images.

Name	Figure	Dimensions	Relative amount of edgels
Bike	3.2	640x480	9.63 %
Car	3.1	640x480	7.31 %
Cow	3.3	500x375	7.89 %
Person1	3.4	640x495	3.78 %
Person2	3.5	360x480	4.70 %

Table 3.1: Basic information on the images.

3.1.1 Images

Five images were chosen manually to conduct the experiments. The Car, Bike and Person2 images are from the Graz02 database, the Cow and Person1 from the Pascal VOC 2009 database [11]. The chosen images are taken in the real world, have low (Person1) to high (Bike) clutter, the objects are of different sizes but are clearly visible.

After converting them to grayscale images the edges are extracted using the Canny edge detector [6]. The five images and their edge images are shown in figures 3.1 - 3.5 and an overview of their basic properties is given in table 3.1.

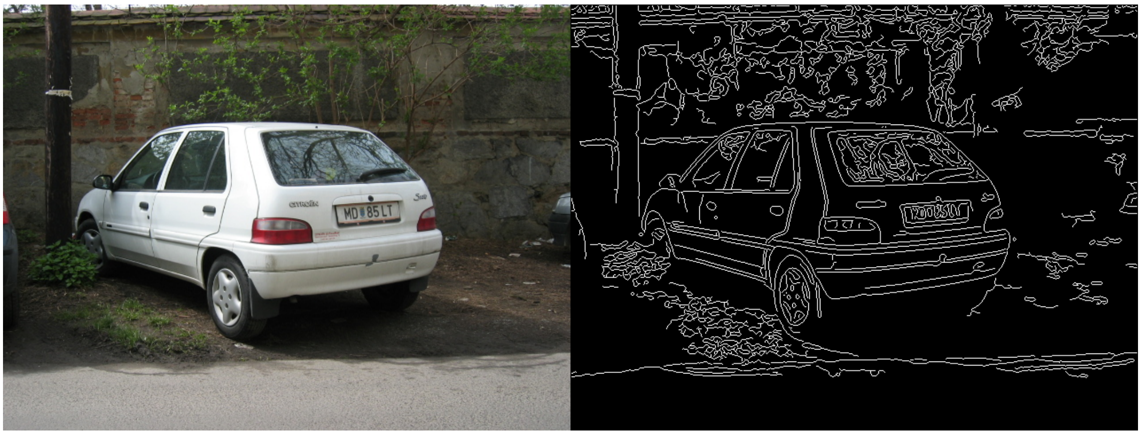

Figure 3.1: Car original and edge image.

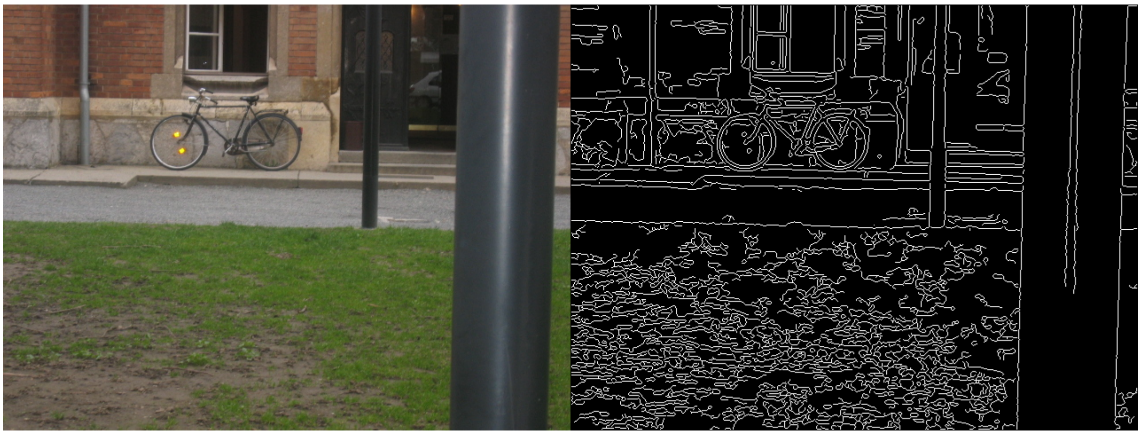

Figure 3.2: Bike original and edge image.

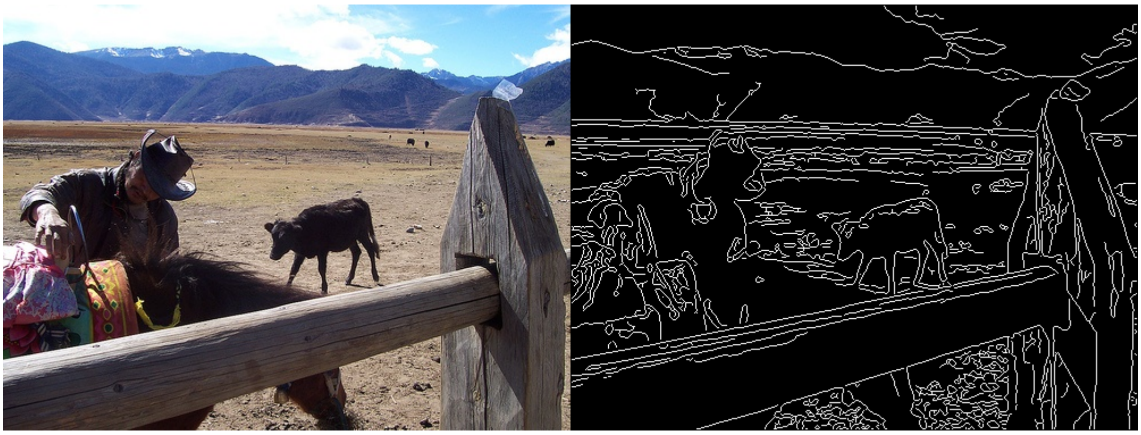

Figure 3.3: Cow original and edge image.

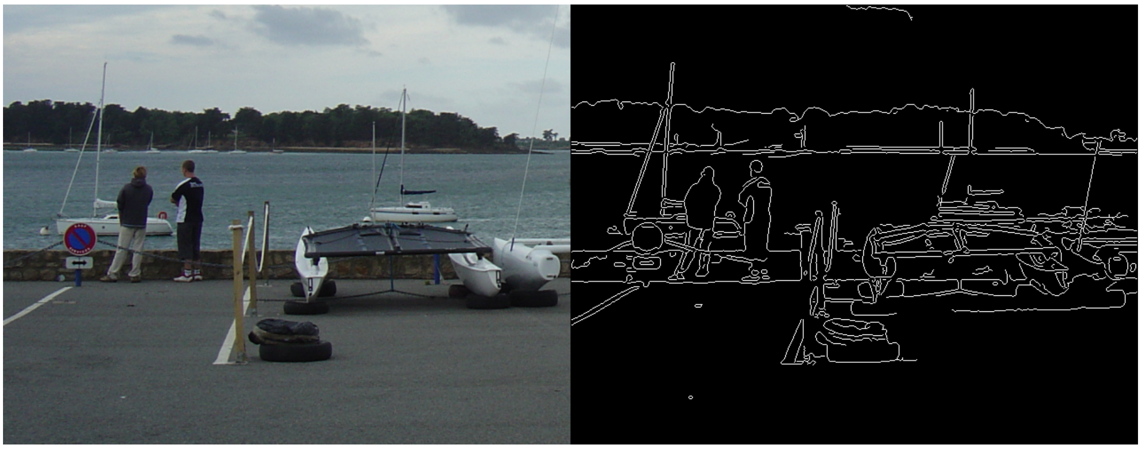

Figure 3.4: Person1 original and edge image.

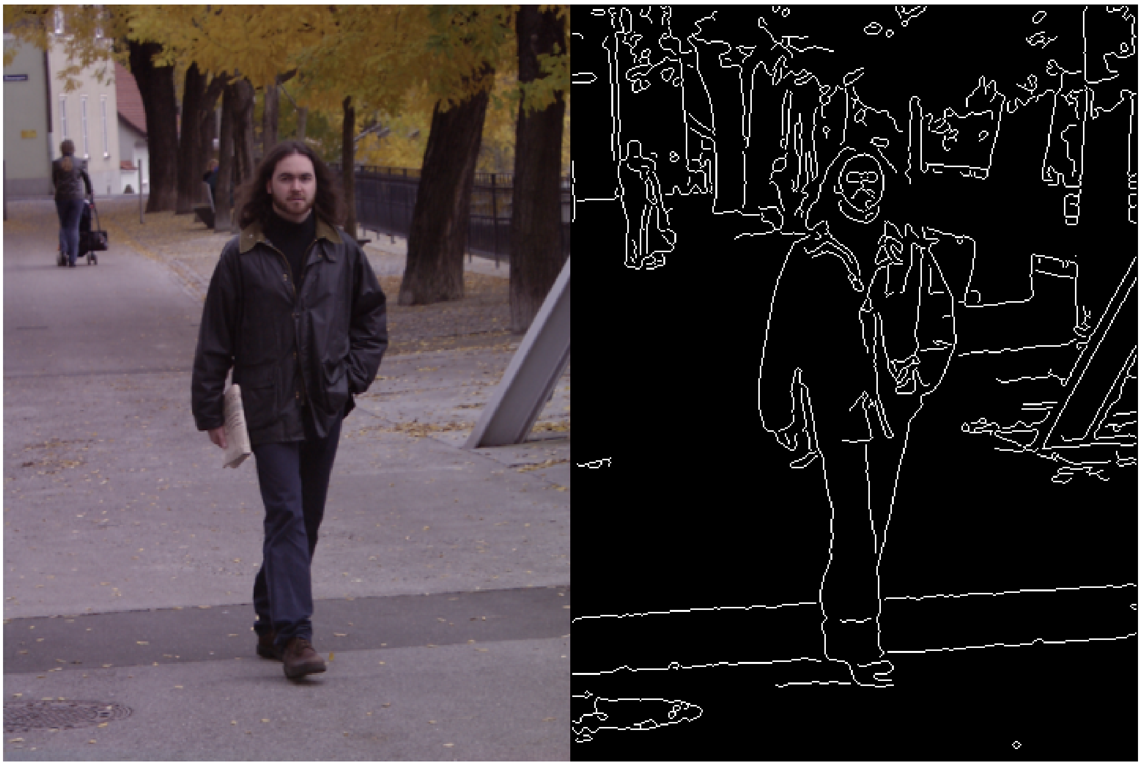

Figure 3.5: Person2 original and edge image.

3.1.2 Contour fragments

Object categories contain many objects, which are similar but not equal. Therefore, the methods from chapter 2 should also be insensitive to small variations. Even if a contour fragment does not fit perfectly, it should have a good quality of match. To test this behaviour, a contour fragment is extracted from an image and then the contour fragment is distorted by an affine transformation. An affine transformation in two dimensions can be parametrised by a scale factor in each dimension, a shearing between the coordinate axis, a rotation and translation. The translation will always set to zero, because the used methods would just find the contour fragment at the translated location. The construction of the transformation is done as follows:

$$T = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} 0 & m \\ 0 & 0 \end{pmatrix}. \tag{3.1}$$

s_x s_y are the scale factors in x and y direction, α is the rotation angle and the shearing will be expressed by $m = \arctan\phi$, ϕ being the angle of shearing.

Two sets of transformed contour fragments are generated: The first set with the parameters

$$\begin{aligned} s_x, s_y &\in \{0.75,\ 1,\ 1.25,\ 1.5\} \\ \alpha, \phi &\in \{-10,\ -5,\ 0,\ +5,\ +10\}, \end{aligned} \tag{3.2}$$

leads to 400 transformed contour fragments. An example set (Cow400) is shown in figure 3.12. The second set using the parameters

$$\begin{aligned} s_x, s_y &\in \{0.75,\ 1,\ 1.25,\} \\ \alpha, \phi &\in \{-5,\ 0,\ +5,\}, \end{aligned} \tag{3.3}$$

leads to 81 transformed contour fragments. An example set (Cow81) is shown in figure 3.11. To denote which set of contour fragments is used for a specific experiment, the amount of contour fragments is added to the experiments. E.g. Cow81 denotes the

Name	Ground truth location
Bike	284, 153
Car	131,134
Cow	301, 208
Person1	151, 183
Person2	120, 147

Table 3.2: Ground truth location of the contour fragments (x,y coordinates).

experiments on the cow image using the set with 81 contour fragments.

The selected contour fragements are shown in figures 3.6 - 3.10, imposed on the edge images in green. The ground truth locations of the contour fragments are given in table 3.2. The upper-left corner of the bounding rectangle to the contour fragment is used as location of the respective contour fragment.

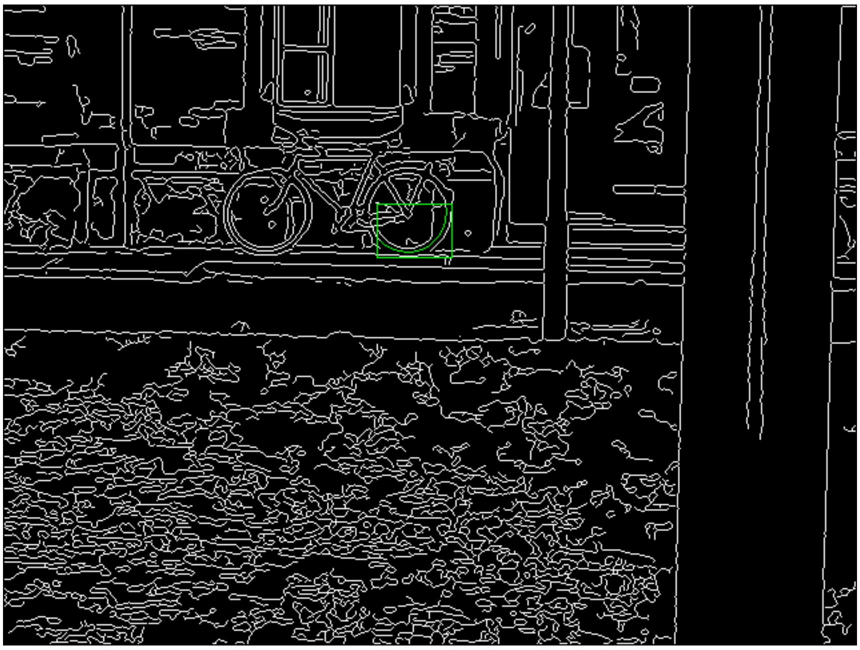

Figure 3.6: The contour fragment (green) selected from the bike image.

3.2 Experiments

Using the methods from chapter 2, 16 methods or combinations of methods were chosen to be tested on the data.

The first group of methods are the methods based on the distance transform and

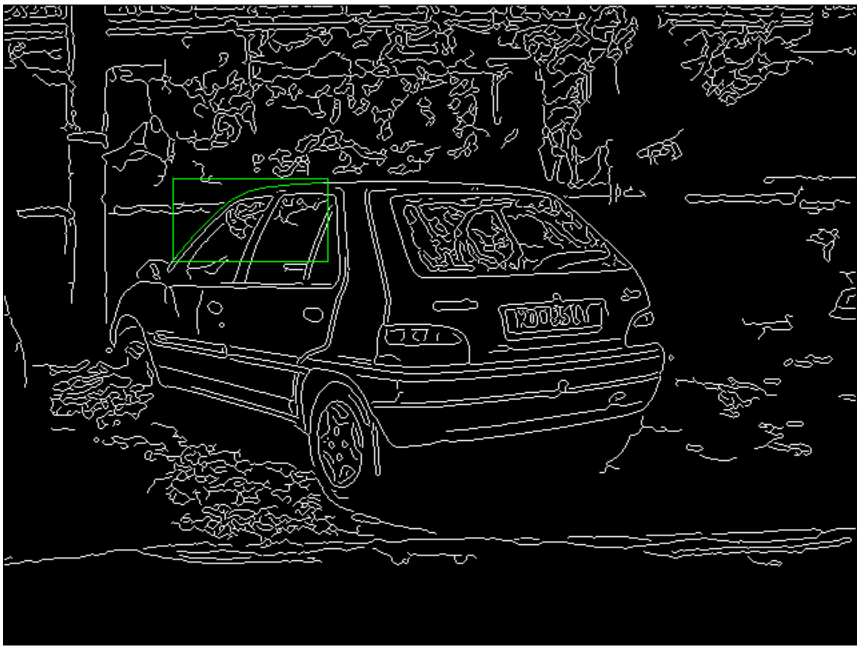

Figure 3.7: The contour fragment (green) selected from the car image.

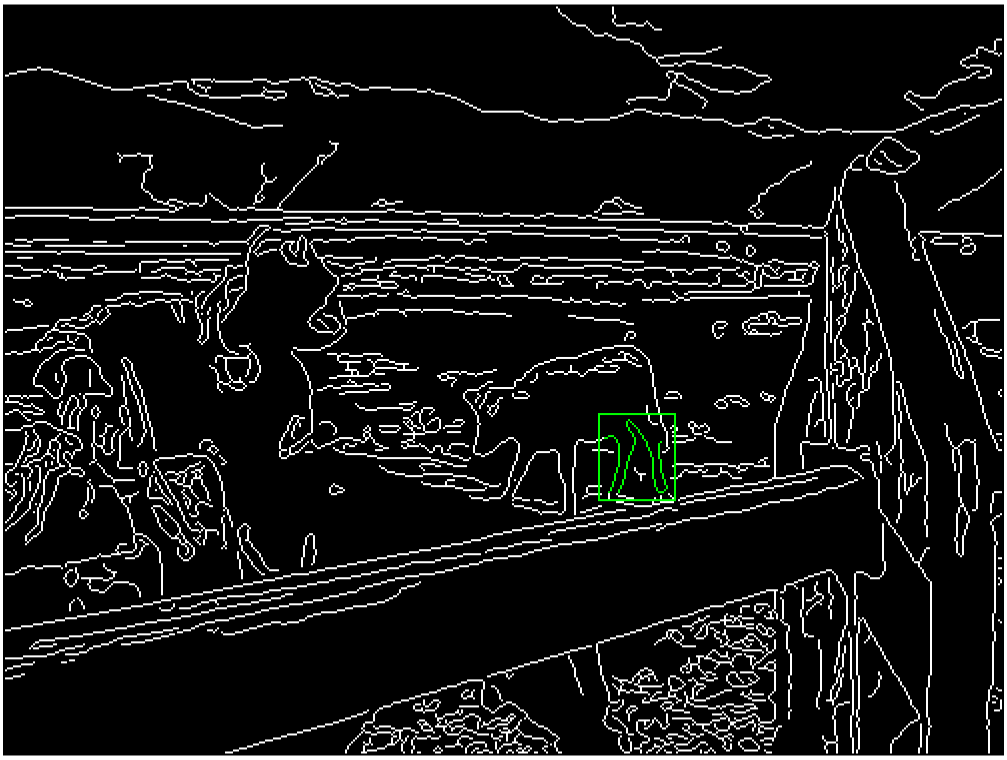

Figure 3.8: The contour fragment (green) selected from the cow image.

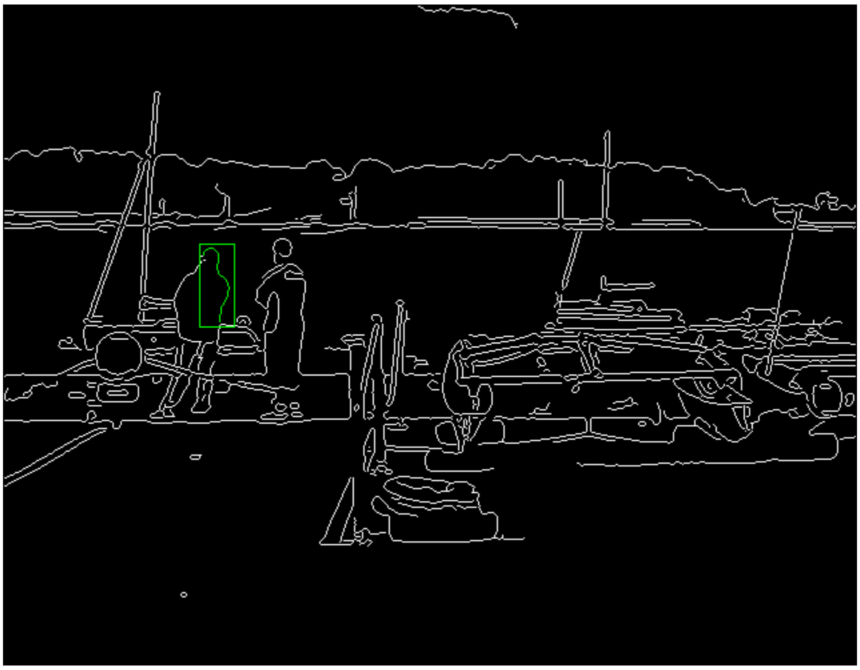

Figure 3.9: The contour fragment (green) selected from the first person image.

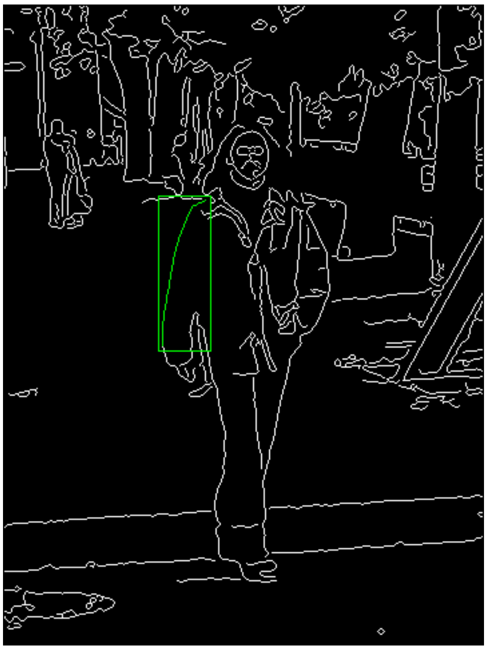

Figure 3.10: The contour fragment (green) selected from the second person image.

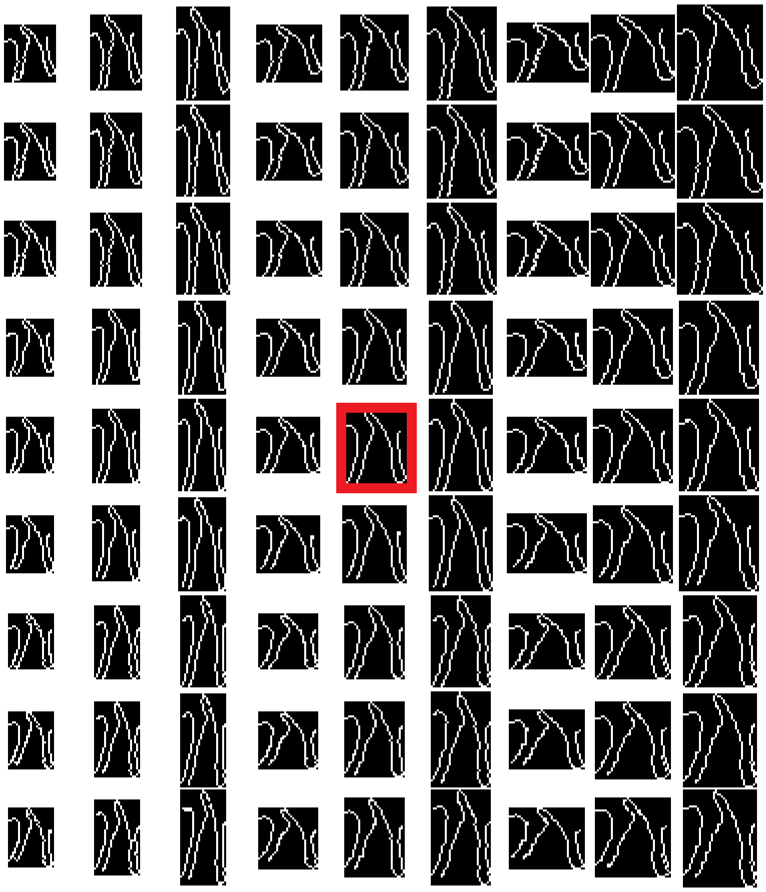

Figure 3.11: The set of 81 contour fragments from the cow image, the red one is the one extracted from the image. In the top - down direction the skew and rotation are changed, in the left - right direction the scale is changed.

correlating the quality image with the contour fragment to obtain a mean distance from the contour fragment to the edge image. Correlating the contour fragment with the distance transform is denoted as “DT” in the remainder of this work. DT is equivalent to the chamfer matching presented by Borgefors [3]. Squaring the distance transfrom before the correlation step will be denoted as “DT2”. Using orientation planes while computing the linear distance transform is denoted as “DT OP”, using orientation planes on the squared distance transfrom as “DT2 OP”. The method suggested by Shotton et al. [26], combining the distance transform with an argument transform is denoted as “DT Shot”.

Using kernel based methods to compute the quality image and then correlating it with the contour fragment are the second group of methods. The standard kernel (see equation 2.7) using $\epsilon = 0.05$ is denoted as “StdKern1” and using $\epsilon = 0.025$ is denoted as “StdKern2”. Combining the standard kernel using $\epsilon = 0.025$ and orientation planes is

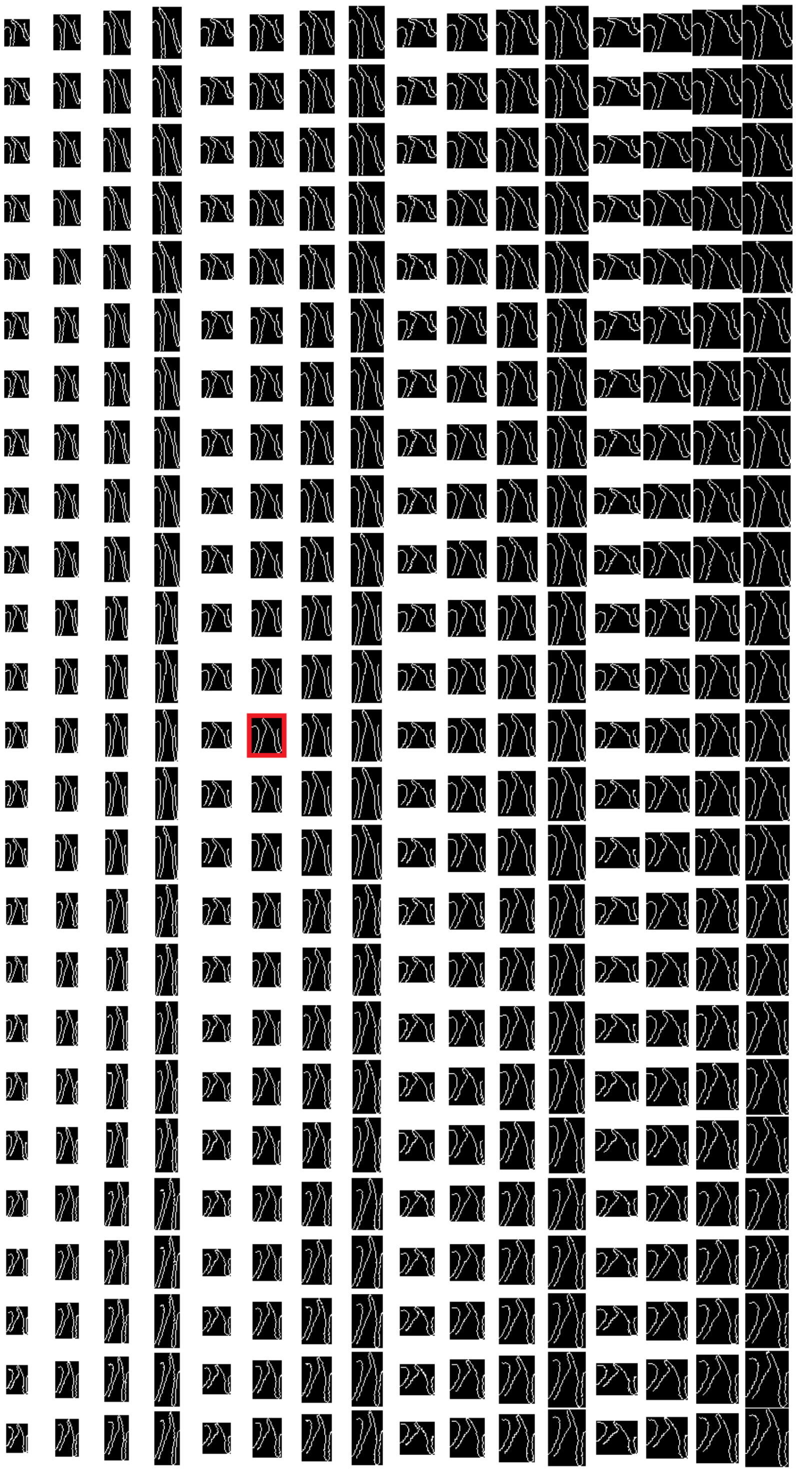

Figure 3.12: The set of 400 contour fragments from the cow image, the red one is the one extracted from the image. In the top - down direction the skew and rotation are changed, in the left - right direction the scale is changed.

denoted as “StdKern2 OP”. Each orientation plane is convolved with the standard kernel and the final similarity image is the sum of all convolutions. Using the oriented Gaussian kernel (see section 2.5.3) is denoted as “GaussKern”.

Computing the variance instead of the mean of the distance transform is the third group of experiments. Two experiments are conducted using only the variance of the distance transform and variance of the distance transform on orientation planes. The former is denoted as “Var” and the latter as “Var OP”.

Another group of experiments is created by combining the DT OP and DT2 OP methods with the variance (Var). The first two experiments (denoted “DT OP Var Add” and “DT2 OP Var Add”) were generated by adding up two results, using a normalizing factor of $\lambda = 0.02$ for the variance part:

$$\text{DT OP Var Add} = \text{DT OP} + \lambda Var \tag{3.4}$$

$$\text{DT2 OP Var Add} = \text{DT2 OP} + \lambda Var \tag{3.5}$$

Another two experiments were generated by multiplying DT OP and DT2 Op with Var, generating “DT OP Var Mult” and “DT2 OP Var Mult”:

$$\text{DT OP Var Mult} = \text{DT OP} * Var \tag{3.6}$$

$$\text{DT2 OP Var Mult} = \text{DT2 OP} * Var \tag{3.7}$$

Finally, the Earth Movers Distance (see section 2.4) is denoted as “EMD”. Due to implementation issues (maximum size of signatures) no experiments were conducted for the Car and Person2 datasets.

Using the above and the two sets of contour fragments for each image, namely the set with 400 contour fragments and the one with 81 contour fragments (as explained in section 3.1.2), a total of 150 experiments is conducted. One experiment tests the combination of a set of contour fragments, the image from which the contour fragment originates, and one of the aforementioned methods.

3.3 Measurements of quality

This section discusses the means to compare the different methods. The main focus is on the properties of the local extrema. As described in section 2.6, only local extrema are used to search hits for a contour fragment in the quality image.

All measures of quality are first evaluated for one of the N results of matching a contour fragment (either $N = 81$ or $N = 400$) to an image and then averaged over all N results. For the WAGE, ExtCnt, B10AvgD and B10MinD this averaging function is the mean, and for Rank and RankNorm the median was used.

3.3.1 Weighted average distance of global extrema to ground truth location (WAGE)

For each match, the location of the global quality extremum $\boldsymbol{e}$ is found and its distance to the ground truth location $\boldsymbol{l_{gt}}$ is computed. The weighted average $\overline{d}$ of these distances is the WAGE. The weight s is the matching score at the respective global extremum and N is the count of matches.

$$\overline{d} = \frac{1}{\sum_{i=1}^{N} s_i} \sum_{i=1}^{N} s_i \left(\boldsymbol{e_i} - \boldsymbol{l_{gt}}\right) \tag{3.8}$$

The lower this measure is, the better is the algorithm. However, in reality not only the globally best match will be used for object categorization. Therefore the weighted average distance between the global extremum and the ground truth location is more of analytic interest.

The weighting should emphasize the global extrema with a good matching score.

3.3.2 Amount of local extrema (ExtCnt)

The amount of local extrema includes every local extremum in the matching score. No critera (e.g. threshold, minimum distance to other extrema) are applied to filter the extrema. A low number of local extrema helps an object categorization algorithm to compute the result faster and more stable.

It has to be said, that a low amount of extrema is not mandatory for a good matching performance.

3.3.3 Average distance of the best ten hits to ground truth location (B10AvgD)

The average distance of the best ten hits to the ground truth location is computed by first finding the ten best hits w.r.t. the matching score in each image, and then computing the mean distance to the ground truth location.

This measure is motivated by Opelt et al. [20], where the ten best matching locations of a boundary fragment are used to test for a weak classifier. Having a lower average distance on the best ten hits means that more contour fragments hit the correct location.

3.3.4 Minimum distance of the best ten hits to ground truth location (B10MinD)

For each match of a contour fragment to an image the hit with the minimal distance to the ground truth location is chosen out of the best ten hits w.r.t. the matching score. The mean of all the values for one experiment is the B10MinD.

If this measure is low, at least one out of the ten best hits w.r.t. the matching score is near the ground truth location. Using this, an object categorization algorithm can use the ten best hits to have a high probability to match a good contour fragment.

3.3.5 Minimum rank of hits within a distance threshold to the ground truth location (Rank)

The measure of the average and minimum distances of the best ten hits (B10AvgD and B10MinD) look at the best ten ranked local minima, and answer the question if it is feasible that the correct location will be found, if just looking at the ten best matches.

To answer the question, how many matches one should look at, the Rank measure is suggested: It uses the local extrema within a certain radius around the ground truth location and returns the rank of the best match w.r.t. the matching score.

To combine the results of the matches of all contour fragments to one value the median is used. The threshold for the radius is $\frac{cf_{diag}}{5}$, where cf_{diag} is the length of the diagonal of the contour fragment.

3.3.6 Normalized minimum rank of hits within a distance threshold to the ground truth location (RankNorm)

As the Rank measure is an absolute measure, it is heavily dependant on the amount of local extrema. The RankNorm measure is the normalized Rank measure using the ExtCnt as normalization factor. This shows the relative amount of local extrema outside the radius threshold with a better matching score than the best one inside.

Chapter 4

Results

In chapter 3, the data and methods used for the experiments were presented, as well as the evaluation of the results. Please refer back to chapter 3 for abbrevations. The detailed results are found in appendix A and in this chapter the results are discussed.

This chapter starts with the discussion of the single measures of quality. This subsection summarizes the results and emphasises results, which are good, bad or unexpected. A comparison between the experiments using 81 contour fragments and the experiments using 400 contour fragments concludes the first part of this chapter

The second part is dedicated to a comparison of different methods and experiments. This is done by ranking the methods w.r.t. to the measures of quality.

4.1 Single results

This section discusses the results of the single measurements of quality. For each measurment of quality, interpretations of the results are given and attention is paid to unexpected or noticable results.

4.1.1 WAGE

The WAGE values are relativly high for both: The set of experiments containing 81 contour fragments and the one containing 400 contour fragments. For the experiment with 81 contour fragments the overall minimal value is 35.58 pixels while the maximum is 259.31 pixels. Using all 400 contour fragments these values change to 74.08 and 304.24 pixels respectively. The main point of these numbers is, that the global extremum in the matching score is rarely hitting the ground truth location.

The really high values for the kernel approaches (StdKern1, StdKern2, GaussKern) in both Bike experiments (table A.8 and A.3) can be explained by the high response to clutter of these methods in the bottom left area of the image. An example is shown in figure 4.1.

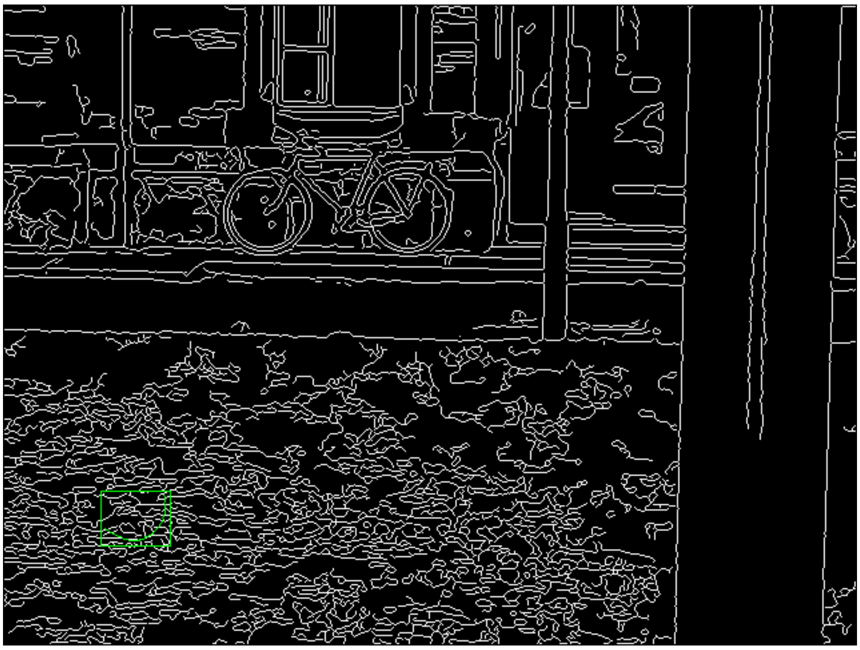

Figure 4.1: Example of a hit in the clutter, using StdKern1.

In the cow experiments (table A.10 and A.5) the methods incorporating orientation planes show poor results. This is due to their hits in the middle left area (textiles on the saddle). An example of one of these hits is shown in figure 4.2.

Matching the same contour fragment to the cow image with the DT method, it is seen (figure 4.3) that it hits in the clutter caused by the dirt in the middle bottom area of the image.

If one takes a closer look at these two clutter regions (figure 4.4), it is seen that the left region contains more vertical edges than the region on the right. The corresponding contour fragment (figure 3.8) also consists of mostly vertical edges.

The best result is achieved by the StdKern2 OP method on the car dataset (table A.4). Only twelve of the 81 contour fragments in this experiment were matching at wrong locations.

Summarizing the results on the WAGE quality measure, it can be said that low values indicate a good algorithm, as seen with the StdKern2 OP method on the car 81 dataset,

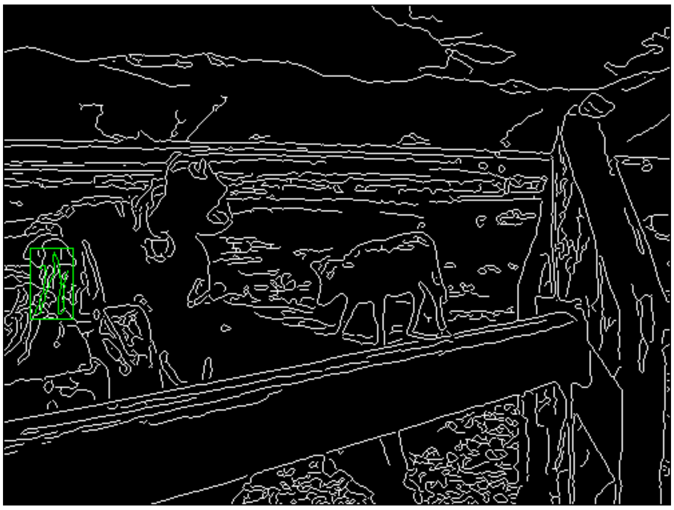

Figure 4.2: Example of a hit in the clutter using DT OP.

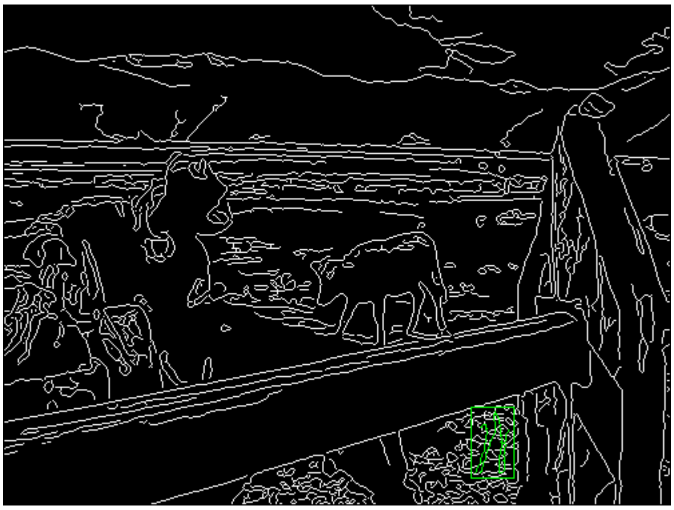

Figure 4.3: Example of a hit in the clutter using DT.

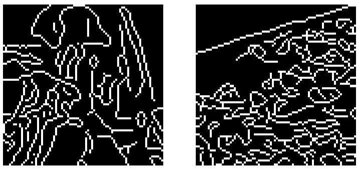

Figure 4.4: Clutter for hits of DT OP on the left, and clutter for hits of DT on the right.

Dataset	Extrema Count
Bike	1550.23
Car	772.65
Cow	650.41
P1	554.07
P2	312.3

Table 4.1: Average Extrema counts on datasets, using 400 contour fragments.

but high values have to be considered with care, because it often depends on the data where a wrong hit finally ends up. This was clearly seen in the comparison between the DT and DT OP method on the cow datasets. If those two regions were to be exchanged, the DT OP method would have scored much better than the DT method.

4.1.2 ExtCnt

The average extrema count depends heavily on the data. The extrema count in the bike dataset is quite high, while it is quite low for the person2 dataset (table 4.1). The extrema count correlates with the amount of edge pixels, the inverse size of the contour fragment and the size of the image.

Almost ten percent of the bike image area are edge pixels and the contour fragment is quite small, whereas the person2 image has a low amount of edge pixels and a large contour fragment.

4.1.3 B10AvgD

Inspired by Opelts method, using the best ten hits, it is analyzed if this contributes to a better result. The first measure to be looked at is the average distance of the best ten hits to the ground truth location. As can be seen in tables A.3 - A.12, the distances are all above 100 pixels. These high values indicate that most of the best ten hits are off target. An example of this is shown in figure 4.5. Most hits, including the best hit, are in the clutter on the left side, while three hits are at some other random location and the correct location is hit once.

4.1.4 B10MinD

To further analyze Opelt´s idea, only the minimal distance of the best ten hits is evaluated. As expected, this measure is much better than the B10AvgD measure. In the example

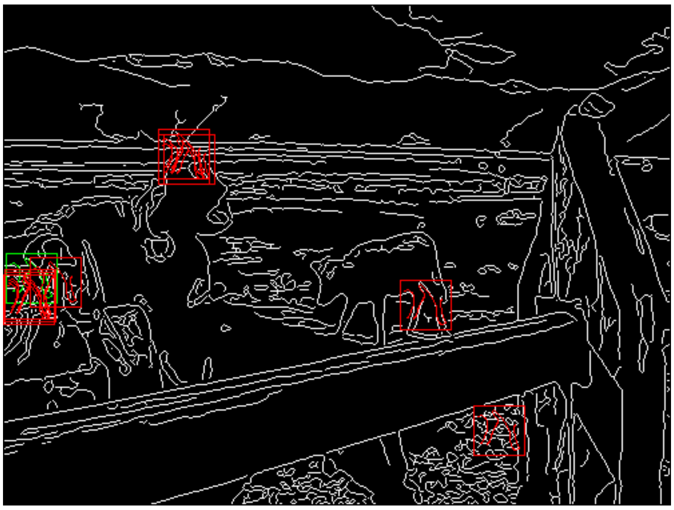

Figure 4.5: Best 10 hits on an image using EMD (best hit is green).

seen in figure 4.5, nine of the then best hits are really far away from the ground truth location, but one hits it nearly perfectly.

The best results in the experiments with 400 contour fragments are a bit above 30 pixels. For large images Opelt accepted a deviation of 30 pixels in the hit accuracy for contour fragments. The methods which get close to 30 pixels in these experiments are already suited for the Boundary Fragment Model.

Looking at the results of the experiments using only 81 contour fragments, many methods achieve results considerably below 30 pixels. These results combined with the results of subsection 4.1.3 show that there are many hits not related to the object the contour fragment stem from, but most of the time one hit is actually hitting the ground truth location. These findings also back up the strategy of getting a good amount of hits (which are mostly false positives) and then try to find the good one by high level processing.

4.1.5 Rank

The best ranked hit within a certain distance is the inverse view on the B10MinD. The results using 400 contour fragments are way above 10 for all images except the car image. Using only 81 contour fragments the better methods are mostly below 10.

These results indicate that for smaller transformations (experiments with 81 contour fragments) the rank threshold 10, chosen by Opelt, is viable. For bigger transformations (400 contour fragments) this is not true and a higher rank threshold should be chosen, if it is still possible to distinguish the good hits from the bad hits by high level processing.

4.1.6 RankNorm

Due to the high variance on the extrema count between various methods, one should look at a normalized version of the best rank. Comparing the methods “DT2 OP Var Add” and “DT OP Var Mult”, on the Person81 experiment, shows the importance of normalizing the rank: While the former has a much better best rank (4 vs 12) comparing the normalized rank reveals that both ranks are around 2.5% of the respective extrema count.

On a more general view it is seen, that the best results from the experiments with 400 contour fragments are ranked within 5% of the extrema count and within 1% of the extrema count for the experiments using 81 contour fragments.

Depending on the method to distinguish the good hits from the bad hits, it could be useful to find a certain relative rank threshold t during the learning phase and then analyze the $t-$best hits to find the object.

4.1.7 Comparision between 81 and 400 contour fragments

The extracted contour fragments were originally transformed using 400 different affine transformations (one being the identity transformation). From these 400 contour fragments a subset of 81 contour fragments is chosen to represent a less severe set of distortions.

The main expectation are better results on the set of 81 contour fragments than on the set of 400 contour fragments. This expectation is met. Furthermore, the results of the experiments on the data sets containing 81 contour fragments suggest that the better methods are robust to the class of transformations used for creating the 81 contour fragments. This can not be said for the experiments using 400 contour fragments.

4.2 Ranking of methods

This section discusses how the different methods perform in relation to each other. After an explanation how to combine multiple results to one ranking, the influence of orientation information is analyzed, whether Opelt´s or Shotton´s method performs better and if squaring the distance increases the quality of the matches. A special look is taken at the EMD and finally the overall rankings are discussed. The detailed rankings of the individual methods are given in tables A.13 to A.24.

4.2.1 Computation of rankings using multiple measures

To compute a ranking from multiple individual measures (e.g. B10MinD and Rank) one could add up the rankings in the individual measures. By doing that, the information how well two methods performed relative to each other is lost. For example, the second best method being just a bit worse than the best method would yield the same result as the best method being way better than the second best.

To preserve this information the result of a method r_i on one particular measure (e.g. WAGE for DT) is normalized by the sum of the results on that measure in that experiment, where N is the count of methods:

$$\overline{r_i} = \frac{r_i}{\sum_{i=1}^{N} r_i} \tag{4.1}$$

Using this approach, all results are comparable and weighted equally. The basis for a ranking using multiple measures is now just the sum of the normalized results $\overline{r_i}$ of these measures.

4.2.2 Influence of orientation information

Both contributions, Opelt et al. [20] and Shotton et al. [26], claimed that information about the orientation of edges improves the localization of contour fragments. In section 4.2.3 it is argued, that in general the orientation planes are the better method to incorporate orientation information. Therefore, the DT and DT OP methods are compared to draw a conclusion about the influence of orientation information.

Comparing the rankings in the experiments using 81 contour fragments, the DT OP method perfoms a bit better then the DT method (table A.29). Taking a closer look at the results, it is seen, that the DT OP performs much better than DT on the Bike, Car and Person1 data set, a bit worse on the Cow dataset and much worse on the Person2 data set (see tables A.25 and A.26).

On the experiments with 400 contour fragments the DT OP method is placed one rank worse than the DT method (table A.30). Again, looking at the detailed results, it is revealed that the Person2 data set is prohibiting a better result of the DT OP method (see tables A.27 and A.28) .

In section 4.2.3 it is also pointed out, that on the Person2 data set, DT Shot performs much better than DT OP. Even if DT OP is not improving the result in each experiment, information about the orientation does.

4.2.3 Orientation planes or argument distance transform?

In the DT OP method, the standard distance transform is augmented by Opelt´s orientation planes, whereas DT Shot augments the standard distance transform by Shotton´s argument distance transform. In both sets of experiments, 81 and 400 contour fragments, DT OP beats DT Shot, by 4 and 3 ranks respectively (see tables A.29 and A.30).

An interesting fact is, that while DT OP performed good on the bike, car, cow and person1 data sets and bad on the person2 data set, DT shot performs good on the person2 data set, and bad on the other four. The seemingly complementary behaviour should be analyzed in a further work, using more images, where DT OP is doing well on one half and DT Shot on the other half. One image is not enough to draw conclusions from.

4.2.4 Squaring the distance

One suggested modification of the standard chamfer matching (DT) is to square the result of the distance transform (see section 2.1.2).

The major advantage is the reduction of the extrema count. The DT2 method has fewer extrema in each experiment and the combination with orientation planes (DT2 OP) leads to the lowest extrema count in all experiments (see tables A.14 - A.20).

In the overall rankings of the experiments with 81 contour fragments (table A.29), DT2 is slightly better then DT (4th rank vs. 6th rank), but this is only due to the better extrema count. Adding orientation planes, DT2 OP is ranked 9th, while DT OP is ranked on 7th place.

Using 400 contour fragments (table A.30), the results are similar, except that DT2 is ranked on second place. Again, the better result in comparison with DT is due to the lower extrema count.

4.2.5 EMD

Due to the high requirements in both memory and computing performance, experiments for the EMD were only conducted for the Cow and Person1 datasets for both 400 and 81 contour fragments, and the Bike dataset for 81 contour fragments.

Despite the computational effort put into the experiments, the suggested method of applying the EMD to the problem of finding contour fragments in images is not suitable. For most experiments the EMD is ranked around place 10, the only exceptions made for the WAGE in the Person1/400 experiment (table A.19, rank one) and the B10AvgD measure in the Person1/81 experiment (table A.15, rank two).

4.2.6 Overall rankings

In tables A.29 and A.30 the final rankings using all measures are seen for the experiments using 81 and 400 contour fragments, respectively.

It is seen that DT OP Var Mult is the best ranked method in both experiments, followed by DT OP Var Add and DT2 OP Var Mult in the experiments using 81 contour fragments, and DT2 and DT OP Var Add in the experiments using 400 contour fragments.

A fact to note is the good ranking of the variance method. Expecting only a mediocre result, because the variance of the distance values only tells how much they change and not the actual values, it is shown by these experiments, that the variance is better than the standard chamfer matching algorithm (DT).

Combining it with the DT2 OP results, it is shown, that the variance of distance values can improve methods which are not so good on their own. Furthermore it is shown, that the variance adds new and valuable information.

The methods based on the distance transform seem to be better suited to tackle the problem at hand. In both sets of experiments these methods are ranked on top, whereas the kernel based methods are ranked at the end of the list. The experiments presented by Dao et al. [9] were done on images with less clutter, wheras the results presented in this work suggest that clutter is a serious problem for kernel based methods.

Also EMD is ranked worse than the distance based methods. Beside the bad results its runtime is another reason not to use it.

Chapter 5

Conclusion

To conclude this master thesis a summary of the presented work is given, as well as a lookout for possible work in the future.

5.1 Summary

In recent years, object categorization algorithms exploiting contour features were presented. Two examples are the works by Opelt et al. [20] and Shotton et al. [26]. Both works rely on the chamfer matching algorithm presented by Borgefors [3] extended by orientation information. Motivated by the lack of analysis about matching contour fragments to images, this work tries to contribute a piece of knowledge about the properties of contour fragment matching.

Starting from the distance transform [3] and the edge potential functions [9], augmentations and modifications are presented which generate a multitude of methods to be evaluated. Some of the modifications are already known in literature (e.g. the orientation planes [20] and argument distance transform [26]), whereas others are newly introduced into the context of matching contours to images (e.g. variance of distances, and oriented gaussian kernel). An adaption of the Earth Mover´s Distance (EMD) is also presented, as it became a popular choice for computing distances in recent years.

To evaluate methods for matching contours to images, new measurements of the quality are introduced. These measurements capture properties of the methods which are related to the success of matching a contour in an image. These measurements are newly introduced and should help to not only compare methods but also to better understand them and their effects on different images.

Finally the most important contribution of this work is the result of the evaluation. It is seen, that the original distance transform performs quite well, but especially the newly introduced idea of not only using the mean but also the variance of the distances improved the matching quality.

The comparison of the B10AvgD and B10MinD measurement shows, that contour fragments probably will always hit in regions of the image unrelated to the object which is searched for. This should be taken into account by methods for object categorization and handled in a proper way.

Clutter is inevitable in real world scenes and it has been shown, that the standard algorithms are prone to it. Using the variance additional to the mean of the distances in the distance transform improves the robustness against clutter, but to completely solve the problem, higher level methods would have to be incorporated.

The evaluation at hand shows the group of methods based on the distance transform clearly ahead, while the adaption of the EMD and the methods based on the edge potential functions are not performing as well. The overall best method in this evaluation is the distance transform augmented by orientation planes and the variance.

5.2 Future work

This work is giving a first look at the problem of matching contour fragments to images. Besides some answers also new questions are found.

Applying a less severe affine transformation to the contour fragments in the data set with 81 contour fragments led to an increase of performance over all methods. It is still to be examined which parameters (scale, rotation, skew) of the affine transformation are influencing the matching result to which degree. Furthermore the robustness of the matching result w.r.t. to the parameters should be analyzed. Knowledge about the behaviour of contour matching algorithms (e.g. slow degradation or abrupt break down), when the contour fragment is not matching perfectly, is of good use when designing a method for object categorization.

While using affine transformations to create distorted contour fragments to test against the images is a controlled and reproducable way creating a test data set, extracting contour fragments from many images of a particular object category could give more insights in to the actual properties of real world objects. While the affine transformation can capture the variants of taking an image of an object, it can not capture the variants of objects within an object category.

The data for the conducted experiments are five images and one contour fragment extracted from each image (see section 3.1 for details). These images were chosen within a context of a general approach. If one has a specific application in mind, the data used to conduct the experiments should be tailored to that application. Not only do object categories have different properties (e.g. rigid vs. non-rigid objects) but also the technical specifications of the imaging system and environmental factors should be considered.

Finally the results of this work should be incorporated into an object categorization algorithm. If multiple methods are implemented, one could find correspondences between the quality of the object categorization method and the quality measures presented in this work. This would render further analysis of methods matching contour fragments to images more precise.

Appendix A

Data

In this appendix the detailed results are given. In subsections A.1 and A.2 the results of the individual experiments are given. A single table gives the measurements of qualitiy for each method for one image-contour set pair. E.g. Bike81 indicates the set of experiments, where the bike image and the corresponding contour set with 81 contour fragments are used.

Subsections A.3 and A.4 show how the different methods are ranked to each other in one measurement of quality. The tables in these subsections allow the comparison of results in one measurement of quality in all images.

Subsection A.5 integrates the results of all measurements of quality per image and shows the rankings of the methods. The method of integration is explained in section 4.2.1.

The final subsection A.6 integrates the results of the single images to an overall ranking. Again using the methods explained in section 4.2.1.

In table A.1 the abbrevations of the methods are explained, the methods are explained in sections 2 and 3.2.

The abbrevations of the measurements of quality are given in table A.2 and explained in section 3.3.

DT	Distance transform
DT OP	Distance transform with orientation planes
DT2	Distance transfrom squared
DT2 OP	Distance transform squared with orientation planes
DT Shot	Distance transform with argument distance transform
StdKern1	Standard kernel 1
StdKern2	Standard kernel 2
StdKern2 OP	Standard kernel 2 with orientation planes
GaussKern	Gaussian kernel
Var	Variance of the distances
Var OP	Variance of the distances with orientation planes
EMD	Earth Mover´s Distance
DT OP Var Add	Sum of DT OP and Var
DT2 OP Var Mult	Product of DT2 OP and Var
DT2 OP Var Add	Sum of DT2 OP and Var
DT OP Var Mult	Product of DT OP and Var

Table A.1: Abbrevations of the methods.

WAGE	Weighted average distance of global extrema to ground truth location
ExtCnt	Amount of local Extrema
B10AvgD	Average distance of the best ten hits to ground truth location
B10MinD	Minimum distance of the best ten hits to ground truth location
Rank	Minimum rank of hits within a distance threshold to the ground truth location
RankNorm	Normalized minimum rank of hits within a distance threshold to the ground truth location

Table A.2: Abbrevations of the measurements of quality.

A.1 Evaluation from 81 contour fragments

This section shows the results of the experiments with 81 contour fragments. Each table corresponds to one image.

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	85,92	2576,8	140,53	13,93	4	0,0016
DT OP	42,48	694,48	124,23	10,65	1	0,0014
DT2	84,64	1632,3	138,36	13,4	4	0,0025
DT2 OP	57,12	339,37	150,88	10,84	2	0,0059
DT Shot	215,04	2609,6	249,31	100,31	42	0,0161
StdKern1	304,96	1668,6	287,87	236,45	471	0,2823
StdKern2	302,96	1292,9	294,17	256,15	625	0,4834
StdKern2 OP	98,16	2425,1	102,15	15,47	3	0,0012
GaussKern	256,72	918,81	253,05	102,48	37	0,0403
Var	83,68	2305,7	139,9	11,45	4	0,0017
Var OP	154,8	2083,9	195,02	37,66	27	0,0130
EMD	140,08	3088,9	193,47	12,04	10	0,0032
DT OP Var Add	44,8	1111,8	117,85	9,73	1	0,0009
DT2 OP Var Mult	58,8	1813,6	103,49	8,53	2	0,0011
DT2 OP Var Add	56,56	412,89	148,91	10,88	2	0,0048
DT OP Var Mult	71,04	1961,4	105,03	8,68	2	0,0010

Table A.3: Results: Bike81

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	67,52	1171,5	120,88	26,17	1	0,0009
DT OP	40,96	520,21	111,89	19,61	1	0,0019
DT2	77,6	535,74	123,43	21,88	1	0,0019
DT2 OP	78,48	274,59	130,88	22,72	1	0,0036
DT Shot	69,68	1193,7	120,6	30,2	1	0,0008
StdKern1	182,8	1004,5	183,49	128,16	47	0,0468
StdKern2	184,32	599,95	190,11	159,05	91	0,1517
StdKern2 OP	47,44	1769,5	127,48	26,31	1	0,0006
GaussKern	195,36	611,99	189,03	78,52	15	0,0245
Var	81,92	862,58	122,57	27,09	1	0,0012
Var OP	98,96	922,99	123,09	34,44	1	0,0011
EMD						
DT OP Var Add	41,84	674,73	107,87	19,76	1	0,0015
DT2 OP Var Mult	69,6	647,73	108,18	25,61	1	0,0015
DT2 OP Var Add	76,96	354,14	128,9	21,64	1	0,0028
DT OP Var Mult	66,48	730,43	112,86	22,71	1	0,0014

Table A.4: Results: Car81

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	112,5	821,46	147,34	62,02	32	0,0390
DT OP	223,44	299,84	234,13	85,71	14	0,0467
DT2	102,19	514,94	151,78	64,78	30	0,0583
DT2 OP	255,06	181,3	234,88	105,14	30	0,1655
DT Shot	118,19	832,07	128,33	93,65	171	0,2055
StdKern1	112,25	389,56	123,78	105,25	124	0,3183
StdKern2	112,25	389,56	123,78	105,25	124	0,3183
StdKern2 OP	155,25	1778,4	155,73	107,07	97	0,0545
GaussKern	113,44	304,49	142,9	104,53	55	0,1806
Var	118,44	925,43	156,54	66,19	18	0,0195
Var OP	235,38	699,14	237,43	128,6	74	0,1058
EMD	259,31	1606,7	241,54	127,86	75	0,0467
DT OP Var Add	198,06	554,27	234	73,55	11	0,0198
DT2 OP Var Mult	197,44	704,23	220,66	67,39	14	0,0199
DT2 OP Var Add	236,81	239,25	239,64	106	30	0,1254
DT OP Var Mult	135,13	804,81	181,27	59,89	11	0,0137

Table A.5: Results: Cow81

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	110,85	583,83	156,34	51,52	96	0,1644
DT OP	107,2	294,72	203,82	17,6	1	0,0034
DT2	122,58	345,95	175,29	49,32	77	0,2226
DT2 OP	173,23	136,33	188,8	24,55	4	0,0293
DT Shot	168,37	601,09	214,7	58,23	176	0,2928
StdKern1	296,29	700,5	245,89	128,09	305	0,4354
StdKern2	267	340,23	265,19	156,72	212	0,6231
StdKern2 OP	191,84	1840,5	201,83	92,15	533	0,2896
GaussKern	175,9	547,51	211,38	60,09	148	0,2703
Var	139,81	573,14	189,83	50,65	33	0,0576
Var OP	204,21	521,84	169,9	36,16	33	0,0632
EMD	138,44	1217	161,2	61,63	208	0,1709
DT OP Var Add	145,72	405,46	196,08	22,4	3	0,0074
DT2 OP Var Mult	205,59	427,06	181,5	33,29	9	0,0211
DT2 OP Var Add	172,5	178,65	188,02	24,58	4	0,0224
DT OP Var Mult	145,07	472,84	180,27	37,43	13	0,0275

Table A.6: Results: Person1 81

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	82,32	273,9	95,78	23,57	3	0,0110
DT OP	104,88	209,57	112,77	61,29	120	0,5726
DT2	73,2	150,22	104,99	25,34	3	0,0200
DT2 OP	104,58	113,11	110,92	59,82	95	0,8399
DT Shot	81,54	288	103,04	24,86	4	0,0139
StdKern1	95,58	613,96	114,09	75,98	137	0,2231
StdKern2	105,3	341,51	105,27	76,78	151	0,4422
StdKern2 OP	120,6	1079,8	126,35	85,96	82	0,0759
GaussKern	125,7	377,54	114,27	37,98	14	0,0371
Var	72,96	292,02	103,71	22,08	2	0,0068
Var OP	102,3	355,01	106,65	67,87	235	0,6619
EMD						
DT OP Var Add	93,54	255,22	107,19	49,19	37	0,1450
DT2 OP Var Mult	90	201,2	101,38	30,67	8	0,0398
DT2 OP Var Add	99,54	139,05	106,46	56,58	81	0,5825
DT OP Var Mult	82,62	243,78	99,85	37,43	3	0,0123

Table A.7: Results: Person2 81

A.2 Evaluation from 400 contour fragments

This section shows the results of the experiments with 400 contour fragments. Each table corresponds to one image.

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	119,2	2748,4	173,03	33,22	56,5	0,0206
DT OP	90,4	701,17	144,19	22,8	187	0,2667
DT2	126,24	1529,3	176,82	38,33	44	0,0288
DT2 OP	95,28	340,57	167,51	28,9	203,5	0,5975
DT Shot	242,64	2509,8	260,41	133,3	327	0,1303
StdKern1	304,24	1661,9	280,4	223,81	719	0,4326
StdKern2	300,64	1259,6	288,24	238,14	656	0,5208
StdKern2 OP	107,6	2463,4	105,77	26,16	14	0,0057
GaussKern	259,68	903,39	261,79	142,24	317,5	0,3515
Var	123,2	2167	180,22	44,25	53	0,0245
Var OP	197,12	1895,7	209,75	69,43	291,5	0,1538
EMD						
DT OP Var Add	79,6	1102,7	140,45	22,08	204,5	0,1855
DT2 OP Var Mult	88	1704,5	133,55	24,8	30	0,0176
DT2 OP Var Add	92,48	416,26	113,2	28,8	211,5	0,5081
DT OP Var Mult	86,72	1849,7	137,65	24,44	21,5	0,0116

Table A.8: Results: Bike 400

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	106,24	1090,1	134,82	57,24	8	0,0073
DT OP	79,76	570,54	124,55	33,99	1	0,0018
DT2	112	478,57	137,95	51,25	7	0,0146
DT2 OP	105,2	282,75	137,51	43,09	4,5	0,0159
DT Shot	102	1111,3	135,76	55,73	9	0,0081
StdKern1	32,96	1290,9	173,34	122,94	65,5	0,0507
StdKern2	139,76	690,63	176,38	142,53	97	0,1405
StdKern2 OP	74,08	2078,9	126,12	30,71	1	0,0005
GaussKern	56,88	741,89	179,83	88,45	25,5	0,0344
Var	116,72	782,94	141	58,37	10,5	0,0134
Var OP	116,32	898,14	132,31	56,9	23,5	0,0262
EMD						
DT OP Var Add	87,84	631,47	123,77	35,31	2	0,0032
DT2 OP Var Mult	108,8	587,86	130,02	54,04	6	0,0102
DT2 OP Var Add	106,16	345,72	135,21	42,12	4	0,0116
DT OP Var Mult	105,84	668,99	131,88	52,41	6	0,0090

Table A.9: Results: Car 400

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	125,13	741,63	140,69	81,27	60,5	0,0816
DT OP	255,25	301,41	242,16	122,34	39	0,1294
DT2	129,5	449,61	143,93	82,65	47	0,1045
DT2 OP	261,25	172,41	238,97	124,22	50,5	0,2929
DT Shot	118,94	754,34	128,89	99,45	146	0,1935
StdKern1	17,06	460,28	132,24	104,32	106	0,2303
StdKern2	17,06	459,87	132,24	104,32	106	0,2305
StdKern2 OP	153,5	1838,5	154,51	110,2	146	0,0794
GaussKern	19,56	328,69	140,9	102,75	57	0,1734
Var	139,13	819,55	147,86	82,28	46,5	0,0567
Var OP	242	651,11	238,89	144,09	105	0,1613
EMD	240,38	1467,8	220,69	126,53	107,5	0,0732
DT OP Var Add	246,63	529,5	245,76	115,87	29	0,0548
DT2 OP Var Mult	236,31	619,8	222,36	94,01	47,5	0,0766
DT2 OP Var Add	254,94	229,43	241,83	126,6	44	0,1918
DT OP Var Mult	173,63	708,81	176,2	79,2	40,5	0,0571

Table A.10: Results: Cow 400

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	140,7	541,8	156,66	56,92	171	0,3156
DT OP	166,03	267,96	191,25	30,76	11	0,0411
DT2	147,66	312,82	170,15	55,54	106	0,3389
DT2 OP	199,44	129,2	171,59	32,57	15	0,1161
DT Shot	185,77	559,53	201,06	64,45	231	0,4128
StdKern1	137,3	7334,9	223,73	111,91	315	0,0429
StdKern2	107,77	4832,4	253,24	128,87	236,5	0,0489
StdKern2 OP	106,07	6823,3	203,4	95,05	528	0,0774
GaussKern	152,19	5798,1	203,41	59,88	165	0,0285
Var	149,03	509,2	183,77	57,84	102,5	0,2013
Var OP	175,9	491,48	159,42	41,39	56	0,1139
EMD	139,97	1163,9	157,84	63,33	278	0,2389
DT OP Var Add	174,28	371,95	173,73	39,06	21	0,0565
DT2 OP Var Mult	204,7	385,16	169,18	49,78	60,5	0,1571
DT2 OP Var Add	199,84	170,29	167,76	33,27	16	0,0940
DT OP Var Mult	180,99	426,27	173,31	51,37	77,5	0,1818

Table A.11: Results: Person1 400

	WAGE	ExtCnt	B10AvgD	B10MinD	Rank	RankNorm
DT	84,9	249,55	98,26	40,43	13	0,0521
DT OP	107,7	200,98	118,15	70,71	132	0,6568
DT2	81,84	134,6	103,44	39,65	22	0,1634
DT2 OP	106,56	103,52	116,76	69,65	103,5	0,9998
DT Shot	86,64	265,22	102,91	41,18	54	0,2036
StdKern1	110,22	593,83	116,6	78,34	181	0,3048
StdKern2	116,76	316,44	110,99	82,74	178	0,5625
StdKern2 OP	126,6	1097,7	128,18	92,43	142	0,1294
GaussKern	128,52	365,84	112,29	49,57	45	0,1230
Var	87,24	263,41	98,82	37,37	16	0,0607
Var OP	105,18	324,27	107,68	73,83	230	0,7093
EMD						
DT OP Var Add	98,76	234,26	106,21	58,52	68,5	0,2924
DT2 OP Var Mult	98,7	184,58	99,78	46,56	54	0,2926
DT2 OP Var Add	102,12	129,1	113,2	66,73	101	0,7823
DT OP Var Mult	94,62	221,14	97,91	40,99	29,5	0,1334

Table A.12: Results: Person2 400

A.3 Individual Rankings for results from 81 contour fragments

This section shows the ranks of the methods for each measurement. This ranks are not to be confused with the measurement Rank. The rankings reflect the results in section A.1.

	Bike	Car	Cow	Person1	Person2
DT	9	5	4	2	4
DT OP	1	1	12	1	12
DT2	8	9	1	3	2
DT2 OP	4	10	15	10	11
DT Shot	13	7	6	8	3
StdKern1	16	13	2	16	8
StdKern2	15	14	2	15	13
StdKern2 OP	10	3	9	12	14
GaussKern	14	15	5	11	15
Var	7	11	7	5	1
Var OP	12	12	13	13	10
EMD	11		16	4	
DT OP Var Add	2	2	11	7	7
DT2 OP Var Mult	5	6	10	14	6
DT2 OP Var Add	3	8	14	9	9
DT OP Var Mult	6	4	8	6	5

Table A.13: Rankings: WAGE, 81 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	14	13	12	12	8
DT OP	3	3	3	3	5
DT2	7	4	7	5	3
DT2 OP	1	1	1	1	1
DT Shot	15	14	13	13	9
StdKern1	8	12	5	14	14
StdKern2	6	5	5	4	11
StdKern2 OP	13	15	16	16	15
GaussKern	4	6	4	10	13
Var	12	10	14	11	10
Var OP	11	11	9	9	12
EMD	16		15	15	
DT OP Var Add	5	8	8	6	7
DT2 OP Var Mult	9	7	10	7	4
DT2 OP Var Add	2	2	2	2	2
DT OP Var Mult	10	9	11	8	6

Table A.14: Rankings: ExtCnt, 81 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	8	6	5	1	1
DT OP	5	3	12	12	12
DT2	6	9	6	4	6
DT2 OP	10	12	13	8	11
DT Shot	12	5	3	14	4
StdKern1	15	13	1	15	13
StdKern2	16	15	1	16	7
StdKern2 OP	1	10	7	11	15
GaussKern	14	14	4	13	14
Var	7	7	8	9	5
Var OP	12	8	14	3	9
EMD	11		16	2	
DT OP Var Add	4	1	11	10	10
DT2 OP Var Mult	2	2	10	6	3
DT2 OP Var Add	9	11	15	7	8
DT OP Var Mult	3	4	9	5	2

Table A.15: Rankings: B10AvgD, 81 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	10	8	2	10	2
DT OP	4	1	7	1	11
DT2	9	4	3	8	4
DT2 OP	5	6	10	3	10
DT Shot	13	11	8	11	3
StdKern1	15	14	11	15	13
StdKern2	16	15	11	16	14
StdKern2 OP	11	9	14	14	15
GaussKern	14	13	9	12	7
Var	7	10	4	9	1
Var OP	12	12	16	6	12
EMD	8		15	13	
DT OP Var Add	3	2	6	2	8
DT2 OP Var Mult	1	7	5	5	5
DT2 OP Var Add	6	3	13	4	9
DT OP Var Mult	2	5	1	7	6

Table A.16: Rankings: B10MinD, 81 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	8	1	9	10	2
DT OP	1	1	3	1	12
DT2	8	1	6	9	2
DT2 OP	3	1	6	3	11
DT Shot	14	1	16	12	5
StdKern1	14	15	14	15	13
StdKern2	15	16	14	14	14
StdKern2 OP	7	1	13	16	10
GaussKern	13	13	10	11	7
Var	8	1	5	7	1
Var OP	12	1	11	7	15
EMD	11		12	13	
DT OP Var Add	1	1	1	2	8
DT2 OP Var Mult	3	1	3	5	6
DT2 OP Var Add	3	1	6	3	9
DT OP Var Mult	3	1	1	6	2

Table A.17: Rankings: Rank, 81 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	6	3	5	9	2
DT OP	5	10	7	1	12
DT2	8	9	9	11	5
DT2 OP	11	12	12	6	15
DT Shot	13	2	14	14	4
StdKern1	15	14	15	15	10
StdKern2	16	15	15	16	11
StdKern2 OP	4	1	8	13	8
GaussKern	14	13	13	12	6
Var	7	5	2	7	1
Var OP	12	4	10	8	14
EMD	9		6	10	
DT OP Var Add	1	7	3	2	9
DT2 OP Var Mult	3	8	4	3	7
DT2 OP Var Add	10	11	11	4	13
DT OP Var Mult	2	6	1	5	3

Table A.18: Rankings: RankNorm, 81 contour fragments

A.4 Individual Rankings for results from 400 contour fragments

This section shows the ranks of the methods for each measurement. This ranks are not to be confused with the measurement Rank. The rankings reflect the results in section A.2.

	Bike	Car	Cow	Person1	Person2
DT	8	8	5	2	2
DT OP	4	2	15	6	11
DT2	10	10	6	3	1
DT2 OP	6	5	16	12	10
DT Shot	12	4	4	10	3
StdKern1	15	14	2	16	12
StdKern2	14	13	2	15	13
StdKern2 OP	7	1	8	11	14
GaussKern	13	15	1	5	15
Var	9	12	7	4	4
Var OP	11	11	12	8	9
EMD			11	1	
DT OP Var Add	1	3	13	7	7
DT2 OP Var Mult	3	9	10	14	6
DT2 OP Var Add	5	7	14	13	8
DT OP Var Mult	2	6	9	9	5

Table A.19: Rankings: WAGE, 400 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	15	13	12	11	8
DT OP	3	4	3	3	5
DT2	7	3	7	4	3
DT2 OP	1	1	1	1	1
DT Shot	14	14	13	13	10
StdKern1	8	12	5	14	14
StdKern2	6	5	5	5	11
StdKern2 OP	13	15	16	16	15
GaussKern	4	7	4	12	13
Var	12	10	14	10	9
Var OP	11	11	10	9	12
EMD			15	15	
DT OP Var Add	5	8	8	6	7
DT2 OP Var Mult	9	6	9	7	4
DT2 OP Var Add	2	2	2	2	2
DT OP Var Mult	10	9	11	8	6

Table A.20: Rankings: ExtCnt, 400 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	8	7	4	1	2
DT OP	6	2	15	11	14
DT2	9	11	6	6	6
DT2 OP	7	10	13	7	13
DT Shot	12	9	1	12	5
StdKern1	14	13	2	15	12
StdKern2	15	14	2	16	9
StdKern2 OP	1	3	8	13	15
GaussKern	13	15	5	14	10
Var	10	12	7	10	3
Var OP	11	6	12	3	8
EMD			10	2	
DT OP Var Add	5	1	16	9	7
DT2 OP Var Mult	3	4	11	5	4
DT2 OP Var Add	2	8	14	4	11
DT OP Var Mult	4	5	9	8	1

Table A.21: Rankings: B10AvgD, 400 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	8	11	2	9	3
DT OP	2	2	12	1	11
DT2	9	6	4	8	2
DT2 OP	7	5	13	2	10
DT Shot	12	9	6	13	5
StdKern1	14	14	8	15	13
StdKern2	15	15	8	16	14
StdKern2 OP	5	1	10	14	15
GaussKern	13	13	7	11	7
Var	10	12	3	10	1
Var OP	11	10	16	5	12
EMD			14	12	
DT OP Var Add	1	3	11	4	8
DT2 OP Var Mult	4	8	5	6	6
DT2 OP Var Add	6	4	15	3	9
DT OP Var Mult	3	7	1	7	4

Table A.22: Rankings: B10MinD, 400 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	6	9	10	11	1
DT OP	7	1	2	1	11
DT2	4	8	6	9	3
DT2 OP	8	5	8	2	10
DT Shot	13	10	15	12	6
StdKern1	15	14	12	15	14
StdKern2	14	15	12	13	13
StdKern2 OP	1	1	15	16	12
GaussKern	12	13	9	10	5
Var	5	11	5	8	2
Var OP	11	12	11	5	15
EMD			14	14	
DT OP Var Add	9	3	1	4	8
DT2 OP Var Mult	3	6	7	6	6
DT2 OP Var Add	10	4	4	3	9
DT OP Var Mult	2	6	3	7	4

Table A.23: Rankings: Rank, 400 contour fragments

	Bike	Car	Cow	Person1	Person2
DT	4	4	7	12	1
DT OP	10	2	9	1	12
DT2	6	10	8	13	6
DT2 OP	15	11	16	5	15
DT Shot	7	5	13	14	7
StdKern1	12	14	14	15	10
StdKern2	14	15	14	16	11
StdKern2 OP	1	1	6	10	4
GaussKern	11	13	11	11	3
Var	5	9	2	8	2
Var OP	8	12	10	4	13
EMD			4	9	
DT OP Var Add	9	3	1	2	8
DT2 OP Var Mult	3	7	5	6	9
DT2 OP Var Add	13	8	12	3	14
DT OP Var Mult	2	6	3	7	5

Table A.24: Rankings: RankNorm, 400 contour fragments

A.5 Ranking using all measures

This section shows the rankings of the methods for each image. The single results are first normalized using the method explained in section 4.2.1 and then integrated by using the mean over all measurements of quality. Tables A.25 and A.27 give the integrated results, tables A.26 and A.28 the corresponding rankings.

	Bike	Car	Cow	Person1	Person2
DT	0.0350	0.0428	0.0444	0.0539	0.0351
DT OP	0.0178	0.0288	0.0484	0.0262	0.0912
DT2	0.0290	0.0350	0.0411	0.0527	0.0316
DT2 OP	0.0194	0.0335	0.0652	0.0297	0.0952
DT Shot	0.0770	0.0444	0.0888	0.0771	0.0368
StdKern1	0.2165	0.1633	0.0851	0.1215	0.0944
StdKern2	0.2780	0.2828	0.0851	0.1228	0.0979
StdKern2 OP	0.0328	0.0492	0.0805	0.1384	0.1005
GaussKern	0.0745	0.0997	0.0590	0.0728	0.0518
Var	0.0326	0.0408	0.0429	0.0448	0.0347
Var OP	0.0507	0.0456	0.0769	0.0444	0.1204
EMD	0.0465		0.0867	0.0796	
DT OP Var Add	0.0199	0.0305	0.0462	0.0314	0.0546
DT2 OP Var Mult	0.0244	0.0350	0.0475	0.0381	0.0378
DT2 OP Var Add	0.0195	0.0334	0.0614	0.0299	0.0803
DT OP Var Mult	0.0264	0.0353	0.0409	0.0367	0.0377

Table A.25: Rankings: Combined results of all measures on datasets, 81 contour fragments.

	Bike	Car	Cow	Person1	Person2
DT	10	9	4	10	3
DT OP	1	1	7	1	10
DT2	7	5	2	9	1
DT2 OP	2	4	10	2	12
DT Shot	13	10	16	12	4
StdKern1	14	14	13	14	11
StdKern2	15	15	13	15	13
StdKern2 OP	9	12	12	16	14
GaussKern	12	13	8	11	7
Var	8	8	3	8	2
Var OP	11	11	11	7	15
EMD			15	13	
DT OP Var Add	4	2	5	4	8
DT2 OP Var Mult	5	6	6	6	6
DT2 OP Var Add	3	3	9	3	9
DT OP Var Mult	6	7	1	5	5

Table A.26: Rankings: Combined rankings of all measures on datasets, 81 contour fragments.

	Bike	Car	Cow	Person1	Person2
DT	0.0476	0.0547	0.0495	0.0622	0.0389
DT OP	0.0467	0.0330	0.0602	0.0333	0.0803
DT2	0.0402	0.0480	0.0452	0.0553	0.0393
DT2 OP	0.0645	0.0421	0.0721	0.0353	0.0833
DT Shot	0.0943	0.0553	0.0709	0.0773	0.0499
StdKern1	0.1426	0.1352	0.0650	0.0990	0.0910
StdKern2	0.1435	0.1940	0.0650	0.1001	0.0896
StdKern2 OP	0.0367	0.0515	0.0843	0.1235	0.1044
GaussKern	0.0963	0.0899	0.0509	0.0655	0.0570
Var	0.0458	0.0560	0.0483	0.0542	0.0397
Var OP	0.0733	0.0701	0.0787	0.0443	0.0974
EMD			0.0827	0.0793	
DT OP Var Add	0.0451	0.0361	0.0560	0.0375	0.0581
DT2 OP Var Mult	0.0327	0.0467	0.0576	0.0481	0.0517
DT2 OP Var Add	0.0574	0.0406	0.0647	0.0351	0.0759
DT OP Var Mult	0.0331	0.0469	0.0490	0.0502	0.0435

Table A.27: Rankings: Combined results of all measures on datasets, 400 contour fragments.

	Bike	Car	Cow	Person1	Person2
DT	8	9	4	10	1
DT OP	7	1	8	1	10
DT2	4	7	1	9	2
DT2 OP	10	4	13	3	11
DT Shot	12	10	12	12	5
StdKern1	14	14	10	14	13
StdKern2	15	15	10	15	12
StdKern2 OP	3	8	16	16	15
GaussKern	13	13	5	11	7
Var	6	11	2	8	3
Var OP	11	12	14	5	14
EMD			15	13	
DT OP Var Add	5	2	6	4	8
DT2 OP Var Mult	1	5	7	6	6
DT2 OP Var Add	9	3	9	2	9
DT OP Var Mult	2	6	3	7	4

Table A.28: Rankings: Combined rankings of all measures on datasets, 400 contour fragments.

A.6 Overall rankings

In this section the overall results and rankings are given. The overall results are the mean of the results in tables A.25 and A.27.

Rank	Method	Result
1	DT OP Var Mult	0.0354
2	DT OP Var Add	0.0365
3	DT2 OP Var Mult	0.0366
4	DT2	0.0379
5	Var	0.0392
7	DT OP	0.0422
6	DT	0.0425
8	DT2 OP Var Add	0.0449
9	DT2 OP	0.0486
10	DT Shot	0.0648
11	Var OP	0.0676
13	GaussKern	0.0710
14	StdKern2 OP	0.0716
12	EMD	0.0803
15	StdKern1	0.1362
16	StdKern2	0.1733

Table A.29: Rankings: Overall rankings and results on the experiments with 81 contour fragments.

Rank	Method	Results
1	DT OP Var Mult	0.0445
2	DT2	0.0456
3	DT OP Var Add	0.0466
4	DT2 OP Var Mult	0.0474
5	Var	0.0488
6	DT	0.0506
7	DT OP	0.0507
8	DT2 OP Var Add	0.0547
9	DT2 OP	0.0595
10	DT Shot	0.0695
11	GaussKern	0.0719
12	Var OP	0.0728
13	StdKern2 OP	0.0801
14	EMD	0.0810
15	StdKern1	0.1066
16	StdKern2	0.1185

Table A.30: Rankings: Overall rankings and results on the experiments with 400 contour fragments.

Bibliography

[1] Barrow, H., Tenenbaum, J., Bolles, R., and Wolf, H. (1977). Parametric correspondence and chamfer matching: Two new techniques for image matching. In *5th International Joint Conference on Artificial Intelligence*.

[2] Belongie, S., Malik, J., and Puzicha, J. (2001). Matching shapes. In *International Conference on Computer Vision*, volume 1, pages 454–461.

[3] Borgefors, G. (1984). An improved version of the chamfer matching algorithm. In *7th International Conference on Pattern Recognition*.

[4] Borgefors, G. (1988). Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(6):849–865.

[5] Breu, H., Gil, J., Kirkpatrick, D., and Werman, M. (1995). Linear time Euclidean distance algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:529–533.

[6] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698.

[7] Chuang, G.-H. and Kuo, C.-C. (1996). Wavelet descriptor of planar curves: Theory and applications. *Image Processing, IEEE Transactions on*, 5(1):56–70.

[8] Dantzig, G. B. (1951). Application of the simplexmethod to a transportation problem. *Activity Analysis of Production and Allocation*, pages 359–373.

[9] Dao, M.-S., De Natale, F. G. B., and Massa, A. (2007). Edge potential functions (epf) and genetic algorithms (ga) for edge-based matching of visual objects. *Multimedia, IEEE Transactions on*, 9(1):120–135.

[10] Dubuisson, M.-P. and Jain, A. (1994). A modified Hausdorff distance for object matching. In *Pattern Recognition, Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 566–568.

[11] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2009). The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results. http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html.

[12] Fidler, S., Berginc, G., and Leonardis, A. (2006). Hierarchical statistical learning of generic parts of object structure. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 182–189.

[13] Fidler, S., Boben, M., and Leonardis, A. (2008). Similarity-based cross-layered hierarchical representation for object categorization. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8.

[14] Fidler, S. and Leonardis, A. (2007). Towards scalable representations of object categories: Learning a hierarchy of parts. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8.

[15] Gavrila, D. (1998). Multi-feature hierarchical template matching using distance transforms. In *International Conference on Pattern Recognition*, volume 1, pages 439–444.

[16] Grauman, K. and Darrell, T. (2004). Fast contour matching using approximate earth mover's distance. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–220 – I–227.

[17] Huttenlocher, D., Klanderman, G., and Rucklidge, W. (1993). Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863.

[18] Kim, W.-Y. and Kim, Y.-S. (2000). A region-based shape descriptor using Zernike moments. *Signal Processing: Image Communication*, 16(1-2):95 – 102.

[19] Lu, G. (1997). Chain code-based shape representation and similarity measure. In *Visual Information Systems*, pages 135–150, London, UK. Springer-Verlag.

[20] Opelt, A., Pinz, A., and Zisserman, A. (2006). A boundary-fragment model for object detection. In *Proceedings of the European Conference on Computer Vision*, volume 2, pages 575–588.

[21] Paglieroni, D. W. (1992). Distance transforms: properties and machine vision applications. *CVGIP: Graph. Models Image Process.*, 54(1):56–74.

[22] Prokop, R. J. and Reeves, A. P. (1992). A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP: Graph. Models Image Process.*, 54(5):438–460.

[23] Revaud, J., Lavoue, G., and Baskurt, A. (2009). Improving Zernike moments comparison for optimal similarity and rotation angle retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):627–636.

[24] Riesenhuber, M. and Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2:1019–1025.

[25] Rubner, Y., Tomasi, C., and Guibas, L. (1998). A metric for distributions with applications to image databases. In *Computer Vision, 1998. Sixth International Conference on*, pages 59–66.

[26] Shotton, J., Blake, A., and Cipolla, R. (2005). Contour-based learning for object detection. In *ICCV05*, pages I: 503–510.

[27] Soille, P. (2003). *Morphological Image Analysis: Principles and Applications*, chapter 6, page 169. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[28] Teague, M. R. (1979). Image analysis via the general theory of moments (a). *Journal of the Optical Society of America (1917-1983)*, 69.

[29] Veltkamp, R. and Hagedoorn, M. (2001). *Principles of Visual Information Retrieval*, chapter State of the Art in Shape Matching, pages 87–119. Springer.

[30] Zhang, D. and Lu, G. (2002). A comparative study of Fourier descriptors for shape representation and retrieval. In *Proc. of 5th Asian Conference on Computer Vision (ACCV)*, pages 646–651. Springer.

[31] Zhang, D. and Lu, G. (2004). Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1 – 19.

[32] Zhang, J.-W., Han, G.-Q., and Wo, Y. (2005). Image registration based on generalized and mean Hausdorff distances. In *Machine Learning and Cybernetics, Proceedings of 2005 International Conference on*, volume 8, pages 5117–5121.