Measuring and Modelling Image Structure

Jon Sporring

Measuring and Modelling Image Structure

Jon Sporring

This is a Ph.D. thesis from the Danish Ph.D. degree in Computer Science at The Department of Computer Science, University of Copenhagen, Denmark. Supervisor has been Peter Johansen, and the advisory board consisted of Mads Nielsen, 3D-Lab, School of Dentistry, Deparment of Paediatric Dentistry, University of Copenhagen, Søren Forchhammer, Department of Telecommunication, Technical University of Denmark, and Mogens Høgh Jensen, Center for Chaos and Turbulence Studies, Niels Bohr Institute, University of Copenhagen.

Copenhagen, January 27, 1999

Contents

Prefa^{ce}

I had a friend in my first school years, who had a little indian tent.

One day we had gotten hold of some cigars which we decided to smoke in the tent. The massive amounts of smoke sent out by a single cigar soon proved to be too much to be contained in the small tent, and we were quickly discovered by the shocked mother of my friend. I don't rembeimber if there was a punishment, but I did'nt have another cigar before I joined the Image group at DIKU.

"So", you may ask, "how did cigars contribute to the science performed?". To this I can truly answer, "not much!". The times when we have been drunk enough to defy once again our female superiors and draw out the cigars, the quality of science has not been high but definitely fun. Thank you for all the cigars.

There are many people that truly deserve a personal remark, but I dare not start on an ever incomplete list, so let me just say: Thank you collaborators and friends. The past three years have been fulfilling.

Nanna and Vibe to you I'm forever in debt.

Jon Sporring

Chapter 1

Measuring and Modelling Image Structure: Introduction

This thesis is a collection of articles written during my Ph.D. study. Some have been published, others are of the status submitted. Some of the articles have been written in collaboration with coauthors, and for this reason have I chosen to include the articles as they were completed in their most final form. I have only taken the liberty to massage their formats to fit this thesis, i.e. rearranging some of the equations and figures, corrected a few errors according to the defense committe, and I have used a single common list of references.

The thesis is organised in two parts. The first part mainly concerns introduction to scale-space and information theory and their uses to solve practical problems. The second part focuses on more theoretical aspects of scale-space and information theory. Each chapter is readable on its own, which I find appropriate for a thesis covering a number of different topics. This also implies that there are more than one introduction to scale-space etc., and that the level and notation is somewhat inconsistent between chapters. To emphasise the general

83	83	86	86	84	83	85	85	85	87	90	94	99	104	109	112
85	86	88	88	86	85	86	86	85	83	82	85	86	92	99	104
85	84	86	87	86	86	87	87	85	86	85	83	83	84	85	90
88	86	85	89	88	85	87	89	90	89	89	89	89	88	87	89
88	87	87	88	90	89	88	88	88	90	88	87	90	88	86	86
86	86	85	89	89	87	88	90	91	90	89	89	90	88	87	87
85	84	82	85	85	86	87	86	86	89	87	86	88	88	86	88
85	85	87	86	86	87	87	88	90	90	90	91	91	88	87	90
86	88	88	89	88	87	88	89	89	88	90	92	91	90	91	92
83	83	85	88	88	88	87	87	87	89	88	90	92	88	87	90
85	85	84	85	86	85	87	90	90	87	89	90	90	89	90	91
86	86	83	87	88	86	87	90	93	91	91	93	94	93	92	91
87	85	86	88	88	86	89	91	91	93	91	92	94	96	93	86
87	84	83	88	91	89	90	92	94	93	94	95	95	93	91	84
87	86	87	87	87	87	89	91	90	91	91	92	94	93	88	76
87	88	91	89	89	89	91	92	92	94	95	95	94	94	90	79

Figure 1.1: A scalar image as seen by the computer: Numbers arranged on a two dimensional grid.

scope of the thesis, three introductory chapters and a final summary have been written. First, below, will be given a very general introduction to image processing and information theory, written in a manner not presupposing expertise in these fields. Then will each of the two parts be preceded by an introduction of the chapters at a slightly more advanced level. Finally, the thesis is ended by an overall discussion of the work and interesting problems that the thesis has raised. We will now introduce image processing.

1.1 A Psychophysical Experiment

We would like to illustrate the task of image processing by a psychophysical experiment. Figure 1.1 shows a small part of an image as seen by a computer. We see that an image is a scalar valued function sampled on a grid. Each point on this grid we call a pixel, and scalar valued images we call gray scale images. Besides these facts, little information is available to guide the human visual system to the contents of this image. In Figure 1.2 is shown a little larger part of the

same image. This time each pixel is represented by a small square of corresponding grayness. We have experienced that given no preknowledge about the image context few have been able to correctly interpret this subimage. The full image is shown in Figure 1.3. The previously presented subimages were taken of the tomato flower in the middle of the image. We observe that after the context of the subimage in Figure 1.2 has been identified, most people will agree that it is now easy to recognise details such as the stem and the leaves etc..

With this example we hope to have emphasized how model dependent the human visual system is, and how effective such a model dependence can be. Details of the subimage cannot be confidently identified before the global model, the tomato, has been identified. With a global model we easily distinguish details such as the leafs and stem, and the details in turn confirm the global model.

Although model driven systems such as the human visual system certainly are powerful, they also have their drawbacks. New features may be very difficult to identify if a model has been selected. We see only what we expect to see.

1.2 Measuring Images

We will now take a closer look at the concept of images. The images considered in this thesis are all the result of a physical measurement process, or at least the simulation of one. The process is illustrated in Figure 1.4. A scene is investigated with a measurement device resulting in a discrete set of points arranged on a grid. This concept of images covers a large range of measurements, such as:

pictures where the measure is on infalling light,

medical images where the measure can be either on the density of protons (magnetic resonance), the human body's shadowing of X-rays (X-ray images or computer tomography) or the echo of ultra-sound, and

statistics where the measure is often on a probability density.

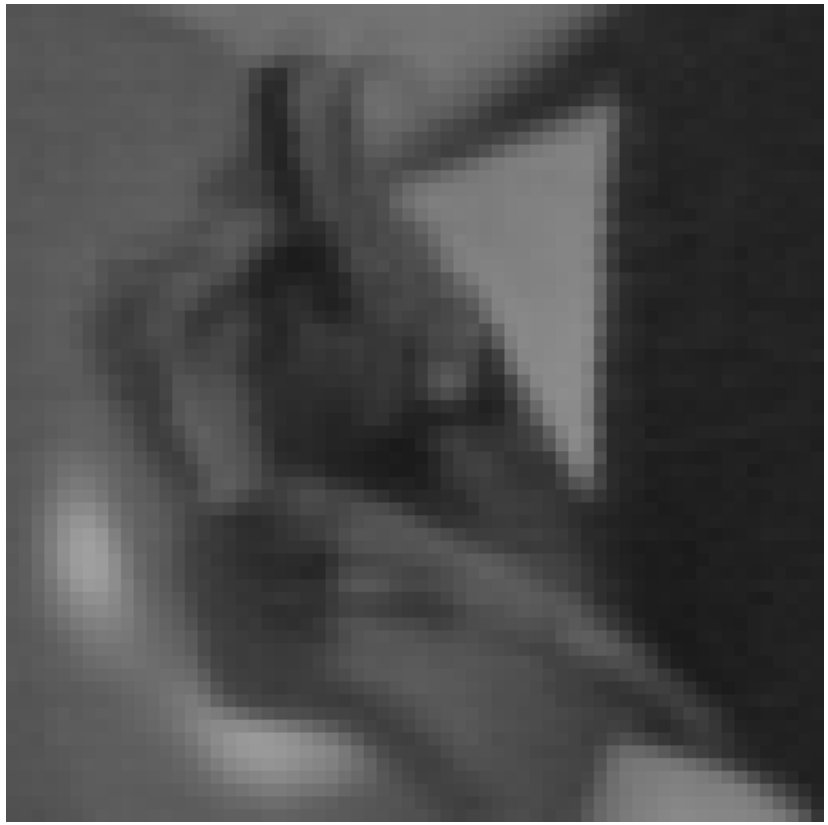


Figure 1.2: A larger subsection of the same image as seen in Figure 1.1. This time each pixel is represented by a small square with corresponding grayness. Even then, the image can be difficult to interpret by humans.

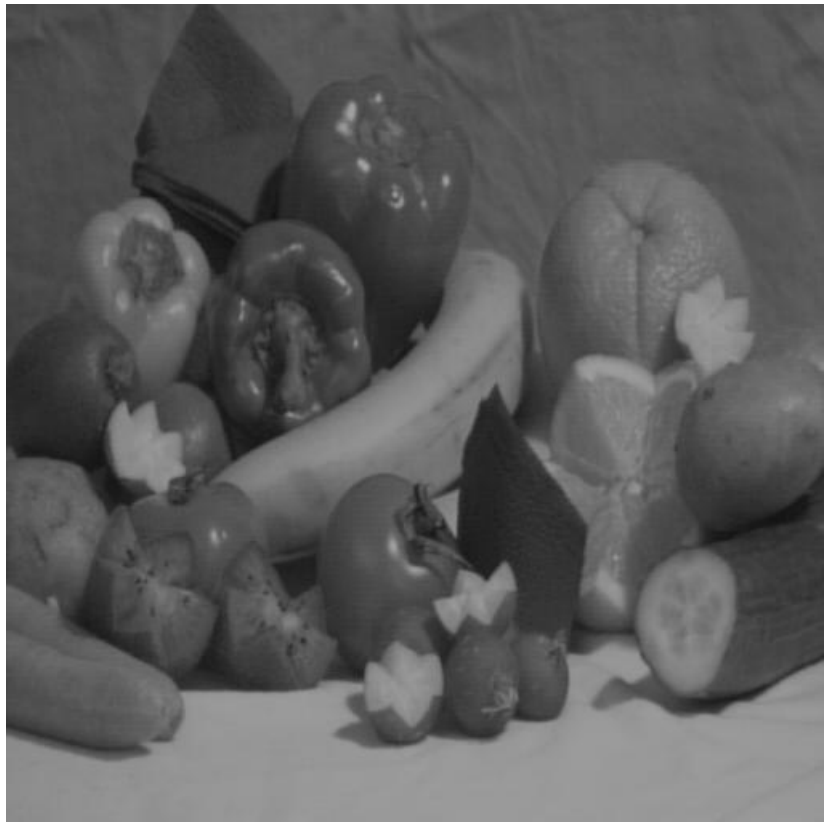


Figure 1.3: The previous two figures (Figures 1.1 and 1.2) showed a zoom of the tomato flower in the middle of this image. Try now to reinterpret Figure 1.2 and note that the task is now much easier.

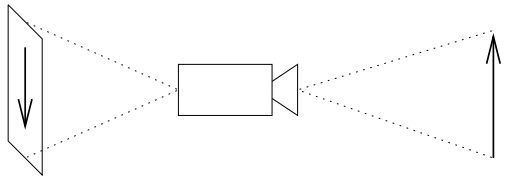


Figure 1.4: The processing of imaging: A measurement device (middle) is used to probe the real world (right) to obtain a discrete image (left).

We call the measured the image modality and the measurement process we call sampling. There is no restriction for the image to be a two dimensional function. One dimensional images are usually called signals, but fit easily into the concept of images. Also three and higher dimensional samples can and will be considered images.

As Figure 1.1 illustrates, each pixel in itself conveys very little information on what is being sampled. In many applications a detailed knowledge of the image modality is of utmost importance, but each image processing algorithm shares a number of common elements. Basic to all algorithms is that they examine the relation between neighbouring pixels. For instance many algorithms will try to estimate derivatives by examining the change of pixel values in prespecified directions, an image may be preprocessed by filtering to reduce noise, or a function may be fitted to the image samples. To give an example, consider the concept of an edge. Examining Figure 1.3 most people will agree that there exists a curve around the tomato which we may call the tomato's edge. In contrast, the zoom in Figure 1.2 shows that the edge of the tomato is not quite so intuitive: Should we include or exclude the tomato-flower? Can we give a precise position of the edge where the tomato overlaps the napkin? We might well find that the edge we choose depends on the contents that we perceive.

It appears that we are at a crossroad, where we may either choose the edge based on local information e.g. by examining the steepness in the transition between light and dark pixels, or on global models such as the conceived tomato. Both approaches will be dealt with in this thesis, but we have experienced that the number and the complexity

of global models is tremendously large in almost all real applications. This is not the case for local models. We are thus in practice dictated a hierarchical approach. For example, by examining each pixel and its immediate neighbours we may assert a likelihood of edgeness. This again can be considered an image, and it will have lines of maximal likelihood. We define these to be the edges. Next step could be to examine the edges to possible identify the tomato. Psychophysical and biological experiments have shown it to be very likely that this is also the overall design of the early stages in the human visual system.

1.3 Reverse Engineering and Data Mining

We identify two trends in the fields of image processing and computer vision: Reverse engineering of the human visual system and data mining. The reverse engineering approach has two merits. Firstly, it is an excellent tool to study and learn human behaviour, and secondly, the human visual system is by far the most versatile and successful image processing system known to date. Data mining is the process of finding hidden patterns and relationships in data. The usual application area is databases, but we find it appropriate to use the term also for image analysis. Specifically, we will use data mining to mean the process of using systems that allow us to learn mostly from data. In this view, a computer is basically a visualisation tool that takes a complicated data set and transforms it into a carefully chosen feature set which can be visualised for human inspections. A simple example is the processing of three dimensional images such as computed tomography images from medical diagnostics. An example is shown in Figure 1.5. It takes years of training for a human to be able to interpret such images by hand. In contrast, a simple technique such as viewing only pixels with a specific value, also known as the isosurface, immediately simplifies this task. In Figure 1.6 is shown an isosurface corresponding to the gray tone of bone from the previous dataset. We see that three dimensional bone structures become much easier to understand once visualised as a surface.

Reverse engineering and data mining have different aims, but re-

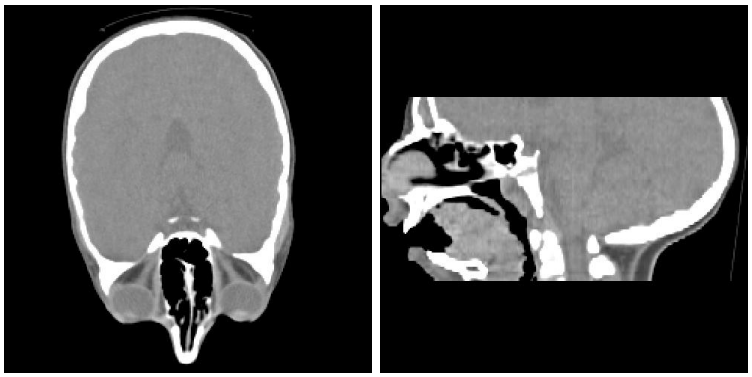


Figure 1.5: Two orthogonal views of a three dimensional dataset. LEFT: A horizontal slice. RIGHT: A vertical slice. The images are courtesy of 3Dlab, Dept. of Pediatric Dentistry, University of Copenhagen, Nørre Allé 20, DK-2200 Copenhagen.

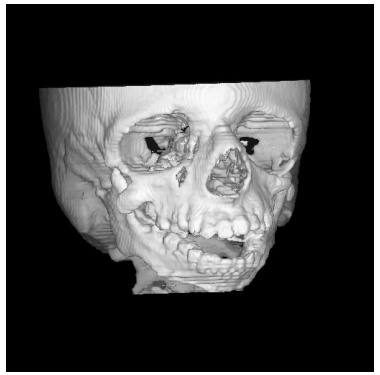


Figure 1.6: An isosurface of the skull also shown in Figure 1.5. The images are courtesy of 3Dlab, Dept. of Pediatric Dentistry, University of Copenhagen, Nørre Allé 20, DK-2200 Copenhagen.

sults from one frequently inspire work in the other. This thesis will emphasise the data mining approach.

1.4 A Guide to the Rest of the Thesis

The main focus of this thesis is on the interplay between image measuring and the modelling of structure. The thesis is organised in two parts. Part I will introduce some key aspects of modelling and measuring image structure. We will introduce the notion of measuring through linear scale-space, spend some time on modelling with differential geometry and the usage of catastrophe theory, and we will end with an introduction to model selection using Information Theory. For this we use two examples: images from a chemical system and images of characters from a fax-document. The second part is more theoretical. In the first two chapters of Part II we take a closer look at gray-value histograms and its sibling from information theory called generalized entropy. An obvious extension is to examine the evolution of histograms, while the function itself is smoothed by scale-space techniques. We will specifically study the mathematical structure of generalized entropies under smoothing transforms, and show how this can be used to select scales in images. Then we will study continuous histograms for one dimensional functions and show that the continuous histograms contain much information about the function itself. The last chapter in Part II we use information theory to examine a model selection algorithm from the field of neural networks called Optimal Brain Damage. We show that the Optimal Brain Damage algorithm has an unspecified but implicit assumption on which neural networks to favour.

Part I

Practical Problems

Chapter 2

Practical Problems: Introduction

Structures in images have a wide variety of sizes, and a general image processing algorithm should be adaptable to the size of structures. An example is edges in images. Coordinates where the intensity change is maximal we call edge points, and the collection of edge points we call edges. In Figure 2.1 are several edges present. This is a 512×512 image, but that is in no way the intrinsic resolution. In fact there does not exist such a thing as the intrinsic resolution for an image. If this image is represented at smaller resolution, e.g. downsampled to a 256×256 image, then certain edges will not be visible. For instance, the feather contains many small edges that are not visible at smaller resolution, while the edge of the hat and the shoulder is still present. This is illustrated in Figure 2.2. A similar result will occur if the photographer was to move away from the object, but in contrast to downsampling moving the camera further away will reveal new parts of the scene at the image border.

Downsampling is a useful operation in image processing, since large structures become small, and small structures disappear. The set of sequentially downsampled images is the so-called image pyramid. The

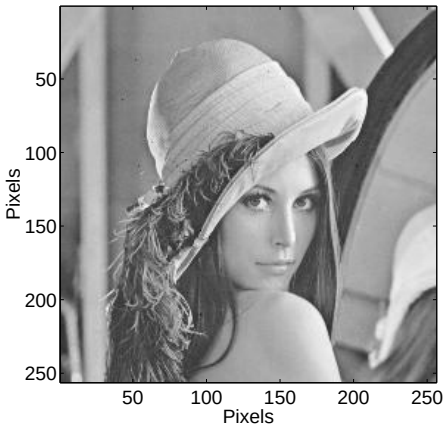


Figure 2.1: An image containing edges at many different scales.

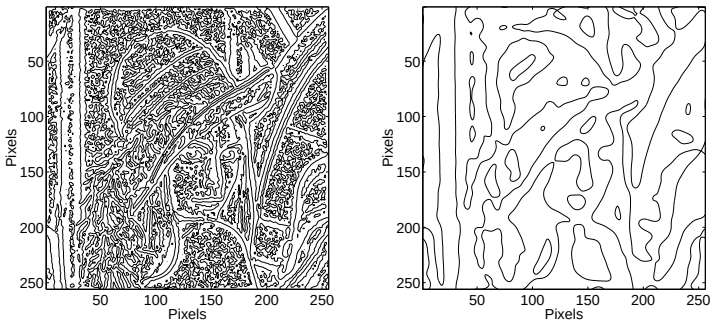


Figure 2.2: The edge images at two different resolutions. LEFT: Edges of Figure 2.1. RIGHT: Edges of Figure 2.1 when the image is downsampled.

advantage of the pyramid representation is that it allows us to design a single algorithm for the analysis of structure at pixel resolution, and reuse it for each image in the pyramid. This is equivalent to writing several algorithms that analyse structures in the original image at sizes 1×1 pixel, 2×2 pixels, 4×4 pixels etc., and apply them to the original image. Although the idea of the pyramid is good, only very few structures reside at integer exponents of the basic downsampling rate. For true scale independent algorithms we therefore seek a tool that allows us to represent an image at any downsampling rate in the least destructive fashion. We shall see that the pyramid is only a crude representation of this tool and the best is found as the convolution with a Gaussian filter, where the standard deviation represents the downsampling factor.

Let us for a moment study the physics of a simple digital camera. A camera is a collection of light sensitive material on a rectangular grid. Each pixel arises from an integration of the infallen light during some time interval also known as the shutter speed. The integration has a positional dependency on the grid, and for mathematical convenience we will assume that the center is often the most sensitive part and the borders the least sensitive[1]. If we assume that the positional dependencies are the same for all pixels, the process of taking an image can be written as the convolution of a filter with the incoming light and sampled on the pixel grid, where the filter corresponds to the positional dependency[2]. The classical image pyramid belongs to this class of algorithms, using a uniform filter corresponding to the downsampling factor.

One may argue that the uniform filter is not the best choice, since an image of an image should not change the image qualitatively. Taking an image of an image is equivalent to performing two consecutive convolutions, and a well known result from mathematics states that

[1] This may not be true for most CCD cameras. Studies have been performed that indicate the border to be the most sensitive part of a CCD pixel cell.

[2] A convolution is the mathematical term for the process of taking local averages. Given a function $f(x)$ and a profile $g(x)$ (also know as a filter, kernel, or distribution), the convolution of the two is defined as: $(f * g)(x) = \int_{-\infty}^{\infty} f(\alpha) g(x - \alpha)\, d\alpha$, where α is a dummy parameter

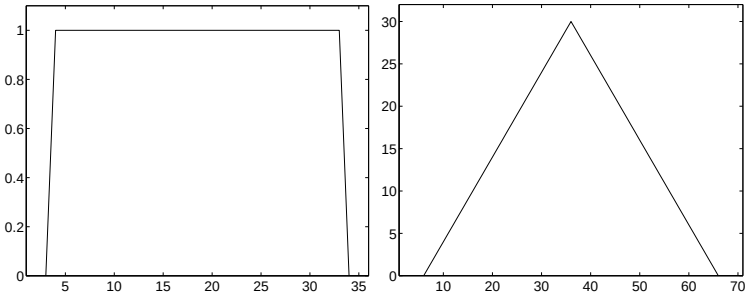


Figure 2.3: The self convolution of a box filter. LEFT: A box filter. RIGHT: The box filter convolved with itself.

two consecutive convolutions can be replaced by a single convolution with a modified filter[3]. Assume that the same camera is used twice, i.e. the same filter is used twice, then the modified filter is found by convolving the filter with itself. A filter will in almost all cases change shape by self filtering. For example, a uniform filter is changed to a triangular filter as shown in Figure 2.3. We may ask, can we now find a camera with a uniform filter that produces the same result as produced by taking an image of an image with a uniform filter. The answer is no. Since self filtering of a uniform filter produces a triangular filter, this will in almost all cases result in a different image than that produced with any uniform filter. For all positive filters having finite variance there is one filter that stays qualitatively the same: The Gaussian filter(Cramér, 1946, p.215). A self filtering of the Gaussian is itself a Gaussian with the double variance. That is, we can easily find a Gaussian camera that produces the identical result as taking an image of an image using a Gaussian filter. In Figure 2.4 is a Gaussian filter shown. Thus, if we use the Gaussian filter in the downsampling process, we need not treat each level in a pyramid as produced by different

[3] Convolution follows the associative rule: $f * (g * h) = (f * g) * h$. Thus if h is the original image, then the consecutive convolutions can be replaced by a single with the filter $f * g$.

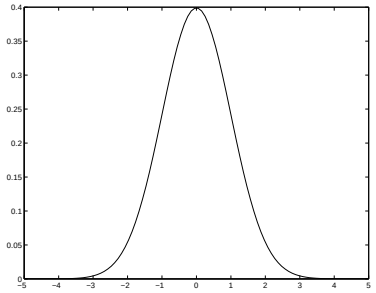


Figure 2.4: A Gaussian function with standard deviation 1.

filters. As a side-remark, note that the sequence of n times self filtering of almost all filters will converge to a Gaussian filter when n goes to infinity. For most algorithms (at least those discussed in the present thesis) it is not necessary to downsample the result of filtering. In this way we get a relatively higher resolution when comparing with the downsampled image. Through these arguments we have reached the Gaussian or linear scale-space. A general introduction to linear scale-space can be found in (Koenderink, 1984; Lindeberg, 1994; Sporring et al., 1997), and detailed review of the numerous axioms leading to linear scale-space can be found in (Weickert et al., 1997a). An introduction to nonlinear scale-spaces that can be used to direct filtering according to image contents can be found in (Weickert, 1998).

Linear scale-space is a useful tool in image processing, and we have applied it to various tasks such as tracking chemical systems and coding blobs in black and white images. We will now shortly introduce Chapters 3–5.

2.1 Tracking Target and Spiral Waves

Differential geometry is a useful tool to define features in images. We have already hinted upon the concept of edges defined through differential geometry, and in Chapter 3 we give a detailed discussion on the problem of tracking spirals and ellipses in a sequence of images.

Taking derivatives is intrinsically an ill-posed problem[4]. Take for example a simple function,

$$f(x) = g(x) - \epsilon \cos(x/\epsilon^2),$$

where ϵ is very small. The second term can be thought of as unnoticeable high frequency noise. But in the derivative of f, the second term becomes as large as ϵ is small.

$$\frac{\partial f(x)}{\partial x} = \frac{\partial g(x)}{\partial x} + \frac{1}{\epsilon}\sin(x/\epsilon^2).$$

In discrete data such noise is always present due to the discretization process. Thus any well-posed discrete differentiation operation should dampen the high frequencies. Gaussian filtering is a very effective method for dampening high frequencies[5]. The calculation of derivatives of Gaussian filtered images is very conveniently done by filtering with the derivative of the Gaussian.

$$\begin{aligned}\frac{\partial^{(i+j)}}{\partial x^i \partial y^i}(f * G)(x,y) &= \frac{\partial^{(i+j)}}{\partial x^i \partial y^i}\int_\infty^\infty \int_\infty^\infty f(\alpha,\beta)G(x-\alpha, y-\beta)\, d\alpha d\beta \\ &= \int_\infty^\infty \int_\infty^\infty f(\alpha,\beta)\frac{\partial^{(i+j)}}{\partial x^i \partial y^i}G(x-\alpha, y-\beta)\, d\alpha d\beta\end{aligned}$$

In Chapter 3 the chemical Belousov-Zhabotinsky reaction is considered. This is a dynamical system which if left alone in many cases will organise itself in spiral and elliptical or target patterns. The spirals and targets seem to originate from a center, and in the experiments considered, these centers move slowly as the reaction progresses. We are concerned with two aspects in relation to image processing: Finding the spiral and target centers and tracking them over time. The solution demonstrates the ease of which linear scale-space can be applied to stabilise differential geometric features.

[4] A problem is ill-posed in the sense of Hadamard (Hadamard, 1902), if the addition of an infinitely small term has an infinitely large effect.

[5] Convolution corresponds to multiplication in the Fourier representation, i.e. $\mathcal{F}[(f * g)(x)] = \mathcal{F}[f(x)]\mathcal{F}[g(x)]$, where $\mathcal{F}$ is the Fourier transform. The Fourier transform of a Gaussian filter is a Gaussian with the inverse variance. Thus the convolution with a Gaussian filter dampens the frequencies exponentially.

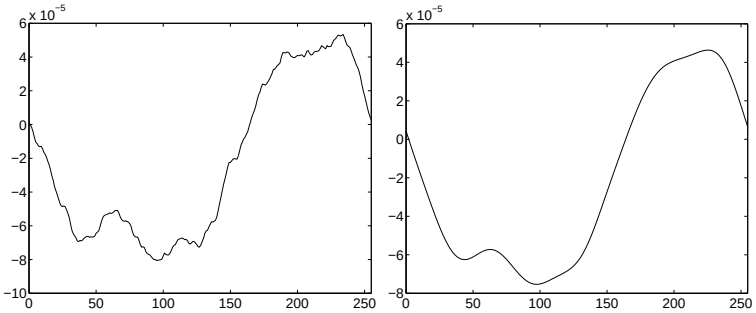


Figure 2.5: Gaussian filtering reduces the number of extrema and dislocates the remaining. LEFT: The original function. RIGHT: The Gaussian filtered function.

2.2 A Note on Differential Corner Measures

Scale-spaces are tools that simplify images. For the linear scale-space on a one dimensional function it can be shown that the number of extrema is non-increasing as scale increases. In Figure 2.5 is shown an example of a one dimensional function before and after filtering with a Gaussian function. We see that the number of extrema is reduced by Gaussian filtering, but also that the remaining extrema are dislocated from their original position. We may illustrate this for the complete set of scales by a fingerprint image as shown in Figure 2.6. This figure has been generated as follows. At each scale the position of the extrema is indicated by points on a line. The collection of lines is the fingerprint image. We see that pairs of extrema have a scale where they annihilate, except for two which will only join at scale infinity. The small extrema annihilate at small scales while large extrema annihilate at large scales. Thus if we wish large scale extrema in the original function, we may analyse the function at large scale and track the position of the extrema to low scale to improve localisation. The tracking is far from always

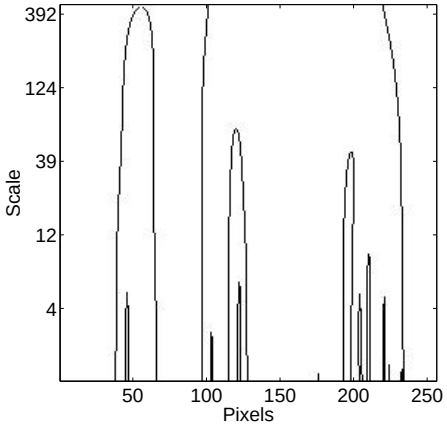


Figure 2.6: The fingerprint image.

as simple as the one dimensional example and often requires a careful study of the deep structure. Chapter 4 studies the deep structure of a small set of corner measures, defined through differential geometry.

The topological studies we will be concerned with in this and the following chapter will be the configuration of singular points. An example is the set of alternating maxima and minima for a one dimensional function. All functions having the same number of extrema belong to either one of two topologically identical function classes: Either the extremum for smallest x is a minimum or maximum. The linear scale-space on a function f has a set of scales where the function changes function classes. These events are called catastrophes. Pictorially, imagine the filming of a piece of wood being bended with increasing force. At some point the wood will break, but if we assume that the breaking takes infinitely short time, this is a catastrophic event, and the exact time of breaking will never be filmed. Always there will be two neighbouring frames: One with the complete piece of wood, and one with two pieces. This is illustrated in Figure 2.7. In our one dimensional example, the bending force is the scale. When the scale is increased, the maxima are reduced and the minima increased. At

Figure 2.7: The filming of a piece of wood breaking.

some scale a pair of neighbouring maxima and minima will annihilate and disappear for almost all functions. This is a catastrophic event since the time for annihilation is infinitely short. We can thus only see a function before and after a catastrophe event, but the effect will be topologically apparent.

In two dimensions, the singular points are defined as the intersection between curves, and Chapter 4 examines corners defined as follows. The curves of constant intensity we call the isophote. All the isophotes will have an isophote curvature, and at some points on these curves there will be extremal curvature. The points of extremal curvature form lines in the image. We define a corner to be the intersection between an isophote and an extremal isophote curvature line. The catastrophe structure is investigated for this and similar corners, and in general we conclude that these measures are both annihilated and created as scale is increased.

2.3 A Piecewise Polynomial Blob Representation

We have in the previous sections seen that linear scale-space can be used to analyse structures of different sizes. We will in Chapter 5 study explicit models of image structure defined via differential geometry and see how linear scale-space can be used to speed up the modelling process.

There is no way around models when we wish to make sense of data, but we cannot consider all possible models. That is, for an image of size $N \times N$ with 256 different pixel values there are 256^{N^2} distinct models that we may consider. Even for relatively small image sizes

this is a very, very large number! If for example the image has size 256×256, the number of distinct models is approximately 10^{157826}. To get a feeling of how huge this number is, imagine we are to search a model space of this size, and we are able to check one model per second. It would then take us about 10^{157819} years to investigate all models. In comparison, the universe is only about 10^{13} years old. This implies that for any realistic task, we only have time to consider a very, very limited number of models. These we call the model class under consideration.

A model from the chosen model class will never fit the image exactly and in a sense need not. What is outside the model class we will call noise and with noise we imply randomness. The randomness can either be a true stochastic source (if such exists), a chaotic process, or the result of a number of minor but complicated processes that are not easily modelled. An example of such a noise process is the electronic noise introduced in cameras during the sampling process. It is important to note, that the concept of noise only has meaning as the dual concept of models; one cannot speak of noise without implying a model since noise implies an error. Conversely, a model always defines the error or noise image. To model images thus implies the investigation of the model and the resulting noise.

Comparing a model with the implied noise is in no way trivial. We will now give a simple artificial example to illustrate this. The example is one dimensional, but the conclusions hold for any image. Consider N data points such as shown in Figure 2.8. These data could for example be temperature measurements taken at the same time and day a number of years in succession. One question that one could ask is, what the general trend in the data is. Is there a rise or fall, and perhaps is the trend accelerating. To answer this, we must choose a model class. The choice should reflect the expectancy we have to the data, i.e. if we suspect periodic patterns then a sinusoidal model class would be appropriate. For the sake of the example we will choose polynomials. The class of polynomials are intimately linked to the notion of derivatives through the Taylor series[6]. In this class

[6]The Taylor series of a function $f(x)$ at $x = a$ is defined as the infinite sum,

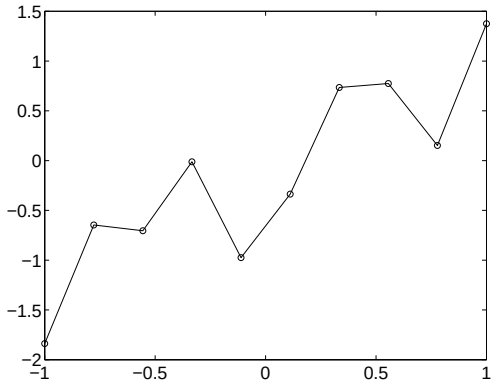


Figure 2.8: A one dimensional dataset.

we have N distinct models that minimise the mean square error[7], the polynomials of order 0 to $N-1$. For the present dataset these are depicted in Figure 2.9. The noise signal is calculated by subtracting the model from the dataset, such that the sum of the model and the noise in the sample points yield the original dataset exactly. I.e. the model and the noise is an exact representation of the dataset. We thus have $N+1$ different representations: The dataset without a model and the N polynomials together with their noise signals. In terms of modelling, the dataset without a model is not very interesting, since nothing is modelled, i.e. no trend is identified. Likewise, the $N-1$ order polynomial is uninteresting since it fits the dataset exactly and thus has no or zero noise signal. The latter situation is identical of having no model. The complete signal is transformed onto another basis, the polynomial coefficients, and no trend is identified. Hence, it is among the remaining $N-1$ models that the interesting models are to be found.

Let us for the moment focus on the noise signal. The concept of

$g(x) = \sum_{n=0}^{\infty} \frac{d^n f(a)}{dx^n} x^n$. If f and g are identical then f is said to be analytical.

[7]The mean square error for a dataset $X = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ and a function $f(x)$ is defined as $\sum_{i=1}^{N} (f(x_i) - y_i)^2 / N$.

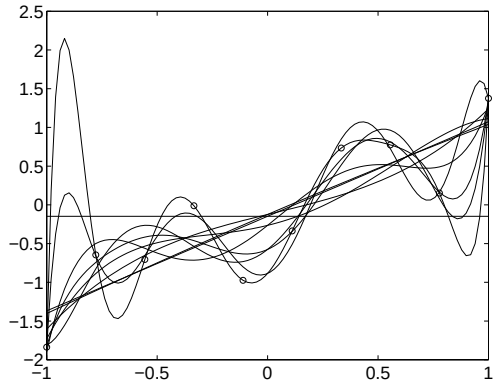


Figure 2.9: The polynomials of order 0 to $N-1$ that minimise the mean square error.

noise is often attributed a statistical meaning in the sense that only the distribution of the signal value can be modelled, not its functionality. While this need not be, we will in this thesis subscribe to this view, since we find it reasonable that if the noise signal has a non-statistical term, i.e. something that we may include in the model class, then this should be included within all reasonable effort. We will thus concentrate on the statistical properties of the noise, and this usually implies the investigation of the mean and the variance[8]. Higher order moments may be investigated, but these are often numerically difficult to estimate. We might also try to estimate a distribution of the statistics of the noise, which again is an example of measuring and modelling as discussed in this thesis. In the present discussion, we will suffice with the studying of the variance, since the mean is assumed to be zero.

It is important to note that the true noise and the estimated or perceived noise are two different entities. Particularly, for all non-artificial signals the true noise is unobtainable! All that can be analysed is the

[8]The mean μ of a discrete source X may be estimated as $\mu = \sum_{i=1}^{N} x_i / N$ where x_i are samples of X and N is large. The variance σ^2 is usually estimated as $\sigma^2 = \sum_{i=1}^{N} (x_i - \mu)^2 / (N-1)$.

difference between a model and the data, the perceived noise. However, given some weak assumptions on the signal, some may be said about the true noise from the perceived noise. In almost all cases, the variance of the perceived noise falls with increasing polynomial order. But as Figure 2.9 shows, even though the perceived noise of high order polynomials has low variance, the functions vary increasingly in between data points and seem very dependent on the particular noise. In this situation it is often said that the data is overfitted, since the stochastic part is sought fitted with a deterministic model. To illustrate this we have performed an experiment as shown in Figure 2.10. An artificial data source has been generated as follows: A line is sampled in ten points and the result is added normal distributed noise with zero mean and unit standard deviation[9]. Several datasets were drawn from this source, and to each we fitted polynomials of various orders. The resulting functions were analysed for mean and variance and plotted in the figure. We see that the standard deviation of the fitted functions is a growing function of order, and in particular that the variance of first order polynomials is smaller than the variance of the noise. A similar experiment is shown in Figure 2.11. Here we have increased the number of sampling points of the source by a factor of ten. We see that when the number of samples is increased, then the variance of the functions also decreases by the same factor.

The statistics of the noise and the degrees of freedom in the model are two different measures that cannot be directly compared. One powerful conversion stems from information theory. The basic idea is that the best description of a dataset is the shortest, which is also sometimes called Occam's razor[10]. The implication is that the elements in the model class are given a unique description of which we may calculate a description length, and likewise for the corresponding noise signals. The theory of descriptions is also called the theory of

[9] A one dimensional normal or Gaussian distribution is given as $G(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$

[10] William of Ockham (1288-1348) did among other things formulate Occam's razor also known as the Law of Parsimony as "Essentia non sunt multiplicanda praeter necessitatem" (Entities should not be multiplied unnecessarily) according to the Catholic Encyclopedia, Electronic Version, 1996.

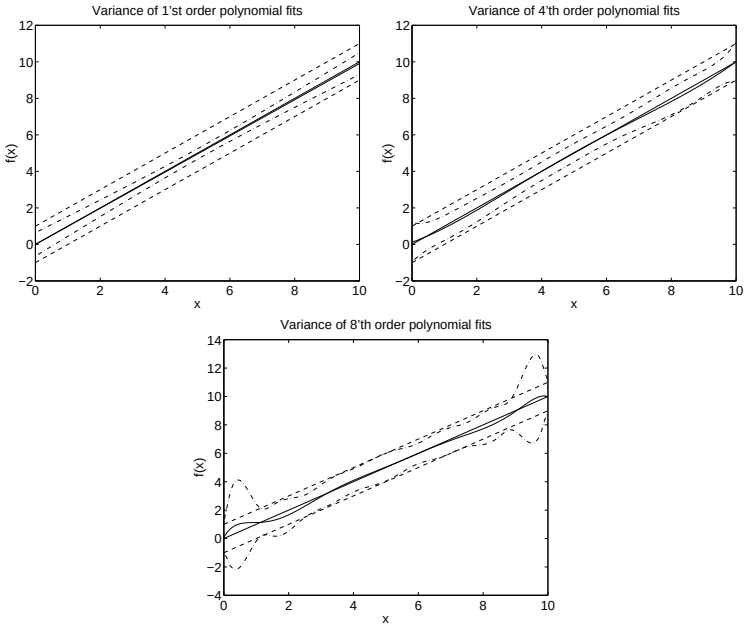


Figure 2.10: The results of the fitting a random 10 point source with polynomials of order 1, 4, and 8. The mean and standard deviation of the source is shown by a solid and two dashed lines. The second solid and two dash-dotted lines show the mean and standard deviation of the fittings.

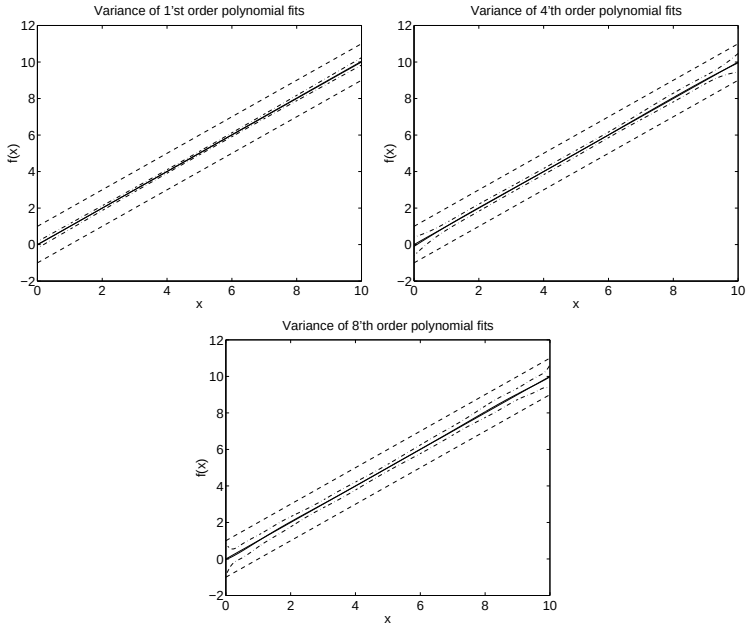


Figure 2.11: A similar experiment as in Figure 2.10. This time the random source is sampled 10 times as densely.

coding, and essentially it implies that both the model class and the noise are assigned a probability. From the probability we then can calculate the lower bound on the description length using a very general class of descriptors, and the lower bound is thus a tool for model selection: That model which has the smallest lower bound is the one to select. In information theory this is called the method of Minimum Description Length (MDL), and in Statistics it is called the Maximum A Posteriori principle (MAP). The reader should note that while these two principles are very similar, only MDL takes direct account for the dependence on the size of the dataset. In our example, we thus have to assign a probability to each polynomial degree and parameter values. This is a hen and egg problem: how can the probability of a model be measured before the model class has been assigned probabilities, and vice versa? This problem becomes even greater if we do not have a number of datasets from the same source, that is, we have no possible way of estimating the probability for the model class. For this reason the probability for the model class is often called the expectancy and is usually set as a subjective choice during the modelling process. In the case where several datasets are available, the description length principle could then be used to refine the expectancy, by choosing the expectancy that minimises the sum of descriptions over all datasets from the same source. In the best cases this is what an experienced data analyst will do in the first case. He will use his intuition to set the expectancy function to reflect the behaviour of the source. But similarly to the psychophysical experiment in Chapter 1, the refinement process relies on the first, subjective expectancy function, and thus new features can be difficult to identify.

Chapter 3

Tracking Target and Spiral Waves[1]

3.1 Introduction

Target and spiral waves in biological, chemical, and physical systems have attracted much attention since the original discovery of such structures in the Belousov-Zhabotinsky (BZ) reaction. Such spatial structures are also observed in convective Rayleigh-Bénard systems, in the aggregating phase of the slime mold *Dictyostelium discoideum* and intercellular Ca^{2+} waves (Cross and Hohenberg, 1993; Field and Burger, 1985; Siegert and Weijer, 1992; Lechleiter and Clapham, 1992). In real experiments the observed patterns usually appear in the form of multiple target and spiral waves separated by more or less sharp boundaries or by regions of spatio-temporal chaos. In order to analyse the long time dynamics of such systems, a huge amount of experimental data must be processed. This type of time consuming analysis is

[1] An earlier version of this chapter has been published as a technical report (Jensen et al., 1998). The current version is submitted for journal publication as: Flemming G. Jensen, Jon Sporring, Mads Nielsen, and Preben Graae Sørensen, "Tracking Target and Spiral Waves".

typically performed after the experiments have been conducted, and it has until now only been performed automatically with methods which depend on the special physics or chemistry of the experiment under investigation (Hanusse et al., 1990; Grill et al., 1996; Winfree et al., 1996).

The method presented in this paper can identify elliptical and spiral waves independently of the mechanism of the pattern forming system. The method is therefore suitable for analysis of a large class of real as well as computer generated patterns. The method combines filtering techniques known as scale-space methods, differential operators on the image level and statistical methods (see (Weickert et al., 1997a) and references therein). The method has time complexity $\mathcal{O}(in \log n)$, where n is the number of pixels and i is the number of images, and it identifies the coordinates of all centers and spiral tips in an image of 256×256 pixels within 40 sec when implemented in Matlab 5.1 on a HP9000s889 running HPUX 10.20.

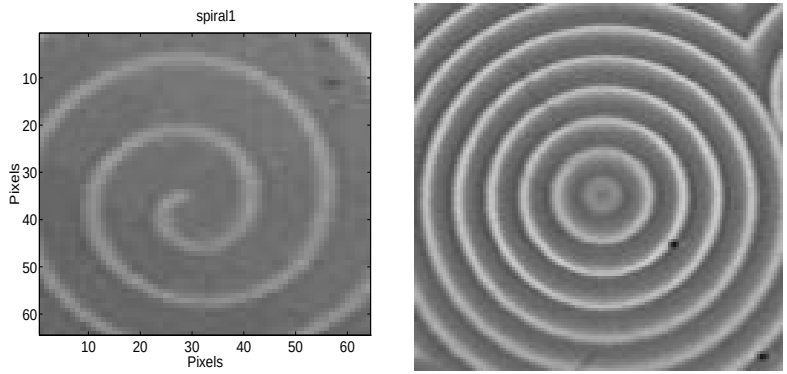
In Section 2 the details of the image processing method are described, and in Section 3 spiral and target centers are traced in 12 BZ experiments catalyzed by the metal complex ruthenium-tris-bipyridyl.

3.2 An Image Processing Approach

In the following we will describe an image processing approach to the analysis of the dynamics of patterns as generated by the Belousov-Zhabotinsky reaction.

The image is an intensity surface sampled on a regular spatial grid (x, y), resulting in a matrix of intensity values $L(x, y)$. In Figure 3.1 are shown images of a target and a spiral pattern from the Belousov-Zhabotinsky reaction.

To study large scale behaviour of spiral patterns Grill *et al.* (Grill et al., 1996) have used the dynamics of the points of constant intensity (loosely speaking the spiral tip). We have used an alternative approach by noting that the evolute of the wave fronts are compactly located close to the center of the spiral and target pattern. We thus propose to define the center of spiral and target patterns to be the center of



(a) (b)

Figure 3.1: An example of a spiral (a) and a target pattern (b) in a Belousov-Zhabotinsky experiment with the initial concentrations $[H_2SO_4]$= 0.8 M, $[BrO_3^-]_0$ = 93.2 mM, $[MA]_0$ = 93.2 mM and $[Ru(bpy)_3^{2+}]_0 = 0.34$ mM.

the evolute of wave fronts. We will use the dynamics of this center to define the dynamics of the spiral and target patterns. An advantage is that a tracking only depends on the speed of the center and n^{ot} on the rotation rate of the spirals.

3.2.1 Calculating the Evolute

We will in the following examine the evolutes of isophotes and edges, where isophotes are curves of constant intensity, and edges are the locus of points of maximal intensity change. By examining the intensity change in the gradient direction we find the edges as the following equation:

$$L_w{}^w \equiv \frac{L_x^2 L_{xx} + L_y^2 L_{yy} + 2L_x L_y L_{xy}}{L_x^2 + L_y^2} = 0. \quad (3.1)$$

The notation introduced in the above formula is a convenient shorthand and will be used in the rest of this article: (x, y) is the Cartesian spatial coordinates, while (w, v) is a local right hand coordinate system, where w is along the image gradient. This is called the gauge coordinate system. Hence, $L_w{}^w$ is the second derivative of L along the w gradient axis. Note that the gauge coordinate system is undefined in extremal points, where the gradient length $L_w = \sqrt{L_x^2 + L_y^2}$ is zero.

The evolute of a two dimensional curve is defined as the locus of points generated by the center of the osculating circle. For a circle the evolute is a point, and for a symmetrical spiral shown in Figure 3.2 (a) the evolute is limited by a circle as shown in Figure 3.2 (b).

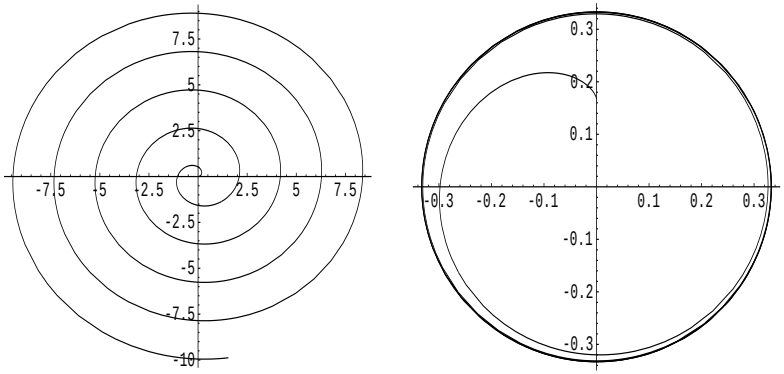
The osculating circle is a geometrical interpretation of the curvature κ of a two dimensional curve: $1/\kappa$ defines the radius of the circle, and the center lies on the line defined by the curve normal N.

For an isophote of an image L, the normal is along the gradient direction

$$\vec{N} = [L_x, L_y]/\sqrt{L_x^2 + L_y^2}, \quad (3.2)$$

and the curvature is calculated as

$$\kappa \equiv \frac{L_v{}^v}{L_w} \equiv \frac{L_x^2 L_{yy} + L_y^2 L_{xx} - 2L_x L_y L_{xy}}{(L_x^2 + L_y^2)^{3/2}}, \quad (3.3)$$



(a) (b)

Figure 3.2: The spiral: $f(s) = [s\cos(s), s\sin(s)]^T$ (a) and its evolute (b). Note that the evolute is limited by a circle.

Due to the discrete nature of images, edges will also be a discrete set of points and therefore also the evolute. Hence we work with the following set:

$$\alpha_k = \{[x, y] + \kappa^{-1}\vec{N} | L(x, y) = k\}. \tag{3.4}$$

The evolute of the $L_{ww} = 0$ edges is estimated in the same manner as above simply by replacing L with L_{ww} in all the above equations and setting $k = 0$, i.e. evaluating on the zero isophote of $L_w{}^w$. For example, the curvature is found by

$$\kappa' \equiv \frac{L_{wwx}^2 L_{wwyy} + L_{wwy}^2 L_{wwxx} - 2L_{wwx}L_{wwy}L_w{}^{wxy}}{(L_{wwx}^2 + L_{wwy}^2)^{3/2}}. \tag{3.5}$$

In this case up to fourth order derivatives are used to extract the curvatures on the edges. Note that $\kappa' \neq L_{wwvv}/L_{www}$ since the $L_w{}^w$ image has a different gauge coordinate system than L.

The image derivatives can conveniently be estimated using Linear Scale-Space (see (Weickert et al., 1997a) and the references therein), i.e. smoothing the image with a Gaussian kernel of standard deviation

$\sqrt{2t}$,

$$L(x, y, t) = G(x, y, t) * L(x, y), \tag{3.6}$$

where the original image is $L(x, y)$ and t is the scale. The advantage of such an embedding is that it reduces the grid and noise effects, allows for a uniform analysis of image structures at all scales, and allows for a well posed estimation of spatial derivatives,

$$L_{x^i y^j}(x, y, t) = G_{x^i y^j}(x, y, t) * L(x, y).$$

In this manner, taking image derivatives of up to fourth order is not an unprecise process for appropriate t (see (Blom et al., 1993; Haar Romeny et al., 1994) for a noise analysis).

We will now demonstrate the difference between the isophotes and the edge approaches on a single image. In Figure 3.3, a single isophote has been shown for the two images together with the corresponding evolute set using Equation 3.3. Immediately we observe that the isophotes are very dependent on the large scale behaviour of the image. The spiral is for example lighter at the top than at the bottom, hence the isophotes can be seen as a dividing line. In this case heuristics must be introduced, and all isophotes with the large gradient lengths are included in the estimate. Still, the drawback of the evolutes of the isophotes is that they are only loosely coupled to the wave fronts.

In contrast as shown in Figure 3.4, we demonstrate the use of the curvature of the edges given by Equation 3.5. As it is seen, the edges follow the stripe structure better and the evolute set is less noisy.

3.2.2 Analysing the Dynamics of the Evolute

Each point on the edge contributes with one point on the evolute. Some of the points will be unrelated to the spiral or the target patterns and may be interpreted as noise in the image, and some will be situated in clusters. For computational reasons, the easiest method of finding the cluster centers is to sample the evolute points on a regular grid, e.g. the same grid as the image is given by, and use Linear Scale-Space to locate the maxima. The scale-space may be applied in two ways: Either by finding extremal points image by image or by stacking the images into

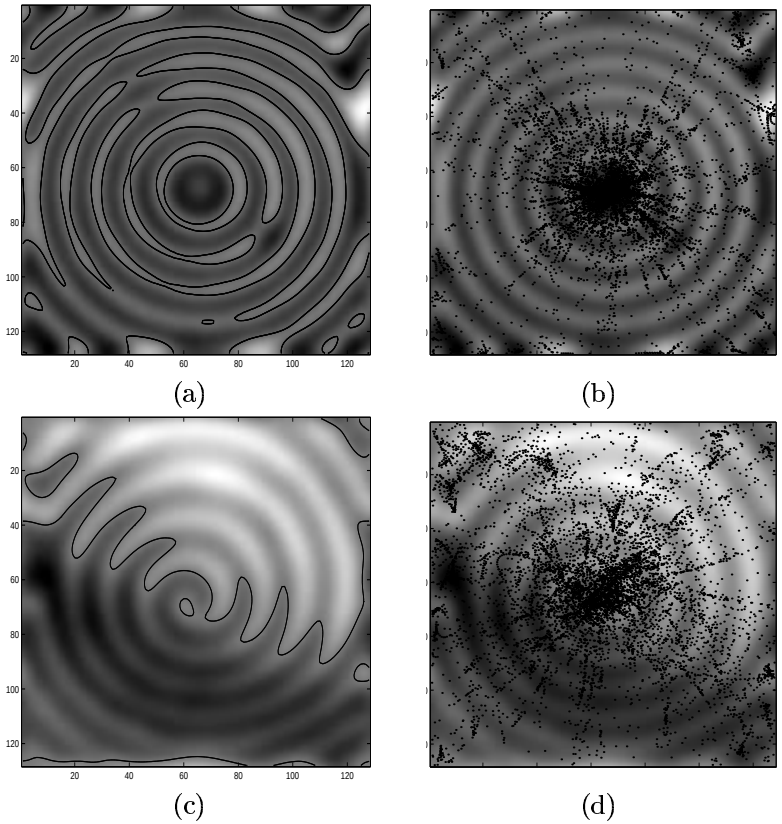


Figure 3.3: The above images show a single isophotes (black lines) for the target pattern (a) and the spiral (c), and the corresponding evolute sets (b) and (d) taken where the gradient is high and at scale $t = 8$.

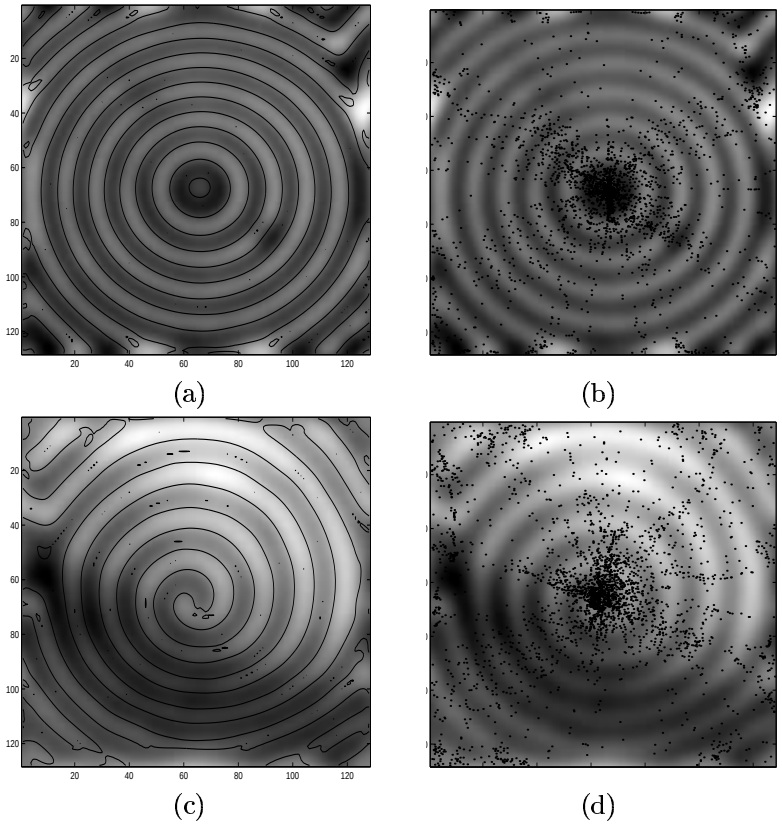


Figure 3.4: The above images show the edge lines ($L_{ww} = 0$) for the target pattern (a) and the spiral (c) and the corresponding evolute sets (b) and (d) taken where the gradient is high and at scale $t = 8$.

a single three dimensional image and locating the three dimensional ridge of extremal points. The latter very reasonably implies that the dynamics is continuous, and it is even possible to implement time causal algorithms that only use past data. But to achieve maximal speed we have chosen to implement the image by image analysis.

To validate the definition of the spiral center, a stable spiral generated by a numerical simulation has been investigated as shown in Figure 3.5. The simulated model is a three variable autocatalator (Peng et al., 1994). The simulation is performed on 256×256 grid using a simple 5 point discrete Laplace operator. The spiral tip performs a cycloid motion as shown in Figure 3.5 (c). In this case the program detects the center of the spiral without the cycloid motion as seen in (b).

3.3 The Experiments

The chemicals used in the experiments were prepared from potassium bromate (Riedel-de Haën 30205), malonic acid (Aldrich M129-6), $Ru(bpy)_3Cl_2$ (Fluka 93307 Tris(2,2'-bipyridyl)ruthenium(II)-chloride Hexahydrate), sulfuric acid (J. T. Baker) and double distilled water. The chloride ions of the catalyst complex were replaced with sulfate ions using a column. The product was tested with a chloride selective electrode.

The experiments were performed in a 9 cm Petri dish and the reaction layer was 0.85 mm thick, i.e. with an aspect ratio $\Gamma_{ext} > 100$. The dish was placed in a thermostated compartment held at 25 ± 0.1 °C, which was purged with N_2 gas to avoid surface reactions between the reactants and O_2 in the atmosphere above the solution. The layer was illuminated from below with a 300 W Xenon arc lamp (Oriel model 66083). The lamp was equipped with a UV grade fused condenser and a photo-feedback system (Oriel model 68850) to obtain a homogeneous distribution of the light on the reaction layer. The light of illumination passed through a central bandwidth filter of 450 $\pm$10 nm. (Spindler & Hoyer). The intensity of the light in the reaction layer was 120 ± 10 nWcm^{-2}. Such low intensity does not perturb the

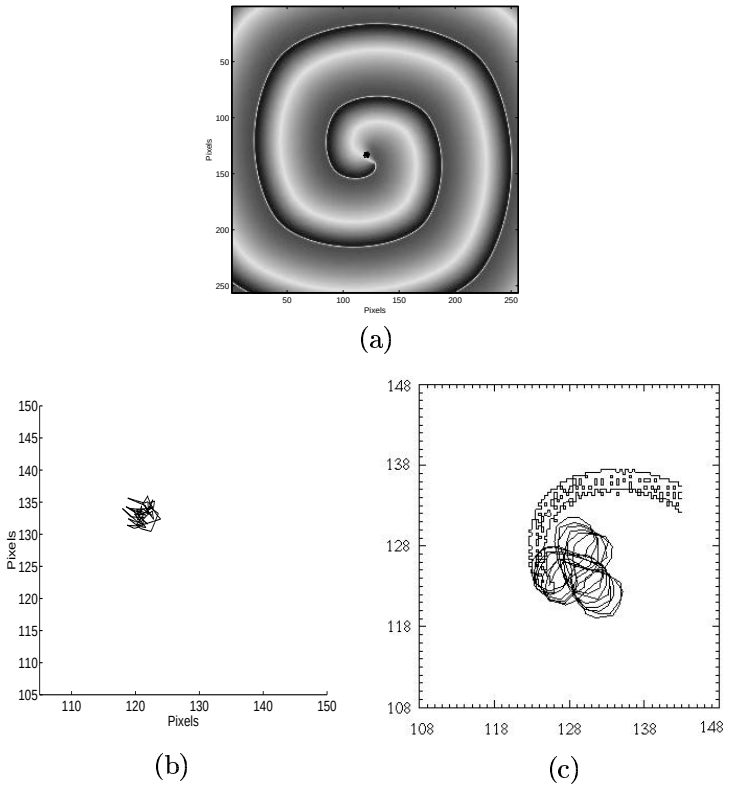


Figure 3.5: Validating the method on a simulated system. (a) shows the image of one component of a simulation of a spiral. The tip is moving. In (b) the detection of the center is shown. No cycloid motion is detected. In (c) a magnification of the motion of the spiral tip is shown. Here cycloid movement is seen.

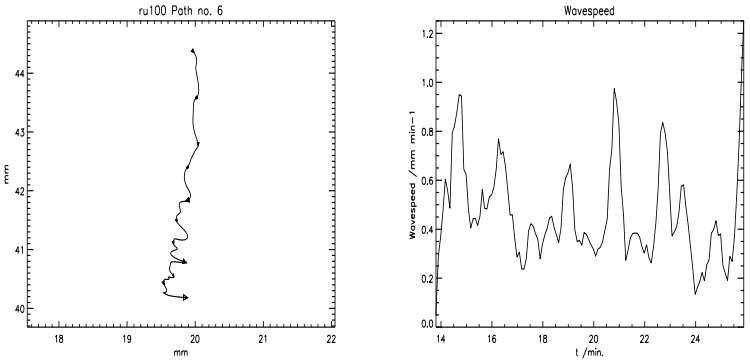
chemistry of the chemical reactions, and in the following we do not consider any interactions between catalyst and light. Nevertheless, we obtained images with high contrast between the oxidised and the reduced areas of the reaction solution. The images of the reaction were captured with a CCD camera (VarioCam PCO CCD) with 720×540 imaging pixels, zoom optics (Fujinon TV-Z 1:18/12.5-75) and a frame-grabber (Imagraph Imascan Chroma-P) before they were stored on a PC. After mixing the chemicals the reaction solution was covered and left undisturbed. Band of travelling waves, oscillating centers and spirals developed spontaneously in the reaction layer, and the evolution of the patterns were monitored until they disappeared, in some cases for more than 1 hour depending on the concentrations of the chemicals.

3.3.1 Tracking Target Centers

The program is able to identify target centers. Waves emitted from such centers are often only visible for short times since target centers are annihilated by travelling waves in the reaction layer because of their lower frequency. Long living target centers in ruthenium catalyzed experiments eventually become distorted from circular to elliptic geometry or even more irregular shapes, which make them difficult to identify. Target centers in the ruthenium catalyzed reaction can move through the reaction solution, as it is seen in Figure 3.6. This center is detected by the program for an interval of 14 min, in which it moves 5.6 mm with a mean speed of 0.39 mm$\,$min^{-1}. The speed of the center is not constant while it was observed. The speed oscillates aperiodic around a mean value, such that the position of the target center is some times almost fixed, while occasionally it moves through the solution.

3.3.2 Tracking Spiral Centers

In the experiments spirals are the dominating spatial structure. In total we have traced the paths of 37 spiral centers in 12 experiments with different values of $[H_2SO_4]$, $[BrO_3^-]_0$ and $[MA]_0$. The initial concentration of the catalyst is fixed to 0.34 mM in all experiments. In



(a) (b)

Figure 3.6: In (a) the path of a target center is shown for an experiment with $[H_2SO_4]$= 0.4 M, $[BrO_3^-]_0 = 93.2$ mM and $[MA]_0 = 93.2$ mM. The arrows indicate the direction of movement. In (b) the mean velocity of the target center is shown as function of time.

Name:	$[H_2SO_4]$ /M	$[BrO_3^-]_0$ /mM	$[MA]_0$ /mM	Length /mm	Time /sec.	$< v >$ sec/mm
$ru233_1$	1.0	93.2	93.2	6.8	59.0	0.11
$ru238_4$	1.0	46.6	93.2	16.6	44.0	0.36
$ru237_1$	1.0	46.6	139.8	7.2	16.0	0.44
$ru202_1$	0.8	139.8	139.8	8.4	29.0	0.28
$ru208_3$	0.8	93.2	93.2	5.6	42.0	0.13
$ru213_4$	0.8	93.2	46.6	12.9	36.0	0.35
$ru213_3$	0.8	93.2	46.6	20.6	46.0	0.45
$ru100_3$	0.4	93.2	93.2	11.0	54.0	0.20
$ru100_4$	0.4	93.2	93.2	11.4	54.0	0.21
$ru101_{10}$	0.4	93.2	46.6	18.6	50.0	0.37
$ru101_9$	0.4	93.2	46.6	11.5	48.0	0.23
$ru119_1$	0.2	233.1	93.2	7.8	23.0	0.33

Table 3.1: The characteristics of 12 spiral centers detected in 9 different experiments. The subscripts on the name of the paths refer to different paths detected in the same experiments.

Table 3.1 the characteristics of 12 paths are listed, together with the different combinations of the chemicals used. The subscript numbers refer to a numbering of the trajectories observed in the same experiment. Note, in this table how centers detected in the same experiment can have different mean speeds.

Figure 3.7 shows trajectories of motion of spiral and target centers in six different experiments. The trajectories are superimposed on images taken halfway through each experiment. The analysis is performed for all the subimages indicated by black squares. The examples were selected to show typical types of motion of the observed patterns. The experiments ru101 Figure 3.7 (a) and ru102 Figure 3.7 (b) are typical and contain mainly spiral centers moving along straight or slightly bended curves. In experiment ru102 Figure 3.7 (b) a big spiral in the center of the image moves perpendicular to the trajectories of the other detected centers to the upper left in the image. In

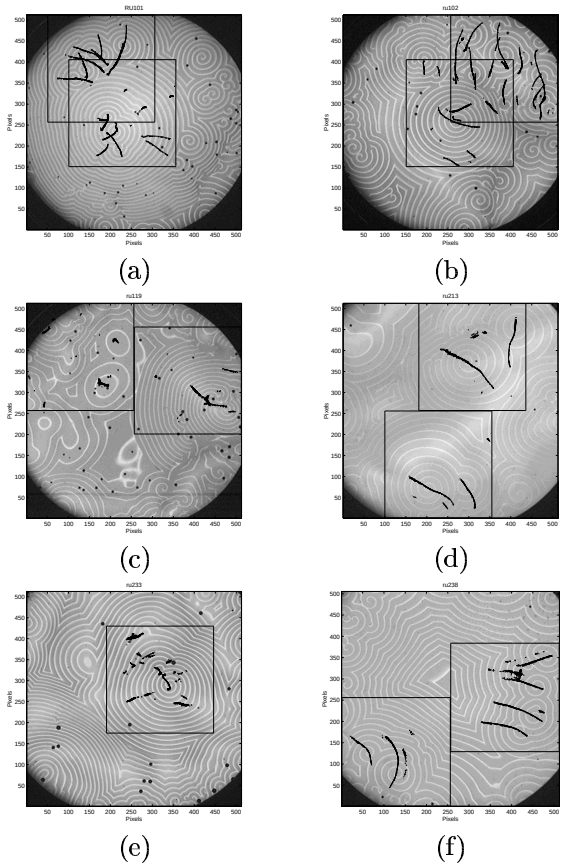


Figure 3.7: Examples of large scale motion of a spiral and target pattern. The movement of the centers are shown superimposed on the midway image. The 256×256 subimage analysed is indicated by a box. Not all paths shown correspond to centers present in the midway image. In general the target patterns are much more unsymmetrical than the spirals. The noisy paths shown in (e) are due to early target patterns.

experiment ru119 Figure 3.7 (c) the reaction becomes turbulent shortly after the time of the shown image, and the spiral dissolves in turbulent waves without any detectable centers. Experiment ru213 Figure 3.7 (d) shows a successful tracking of centers in an image with very low contrast between the oxidised and the reduced parts of the reaction solution. Experiment ru233 Figure 3.7 (e) demonstrates a complicated s-shaped movement of a spiral center. In this experiment the noisy paths are due to some early very elongated target patterns, for which it is difficult to define a true center. The experiment ru238 Figure 3.7 (f) shows bended as well as almost straight trajectories within the same experiment.

For the spiral centers we find, that the mean speed of different centers observed in the same experiment can vary more than a factor 1.5, see Table 1. In all experiments the spiral centers move, but characteristics such as the length of the paths, the observation time and the mean speed of the centers are different. Under our experimental conditions we have not been able to relate the mean speed of a spiral center to $[H_2SO_4]$, $[BrO_3^-]_0$ or $[MA]_0$. This property of spiral centers is in contrast to the empirical relation found by Ram Reddy *et al.* (Reddy et al., 1994) for the velocity of travelling waves: $v \propto \sqrt{[H_2SO_4][BrO_3^-]}$. As an example we find the fastest moving spiral center has $< v >= 0.45$ mm min^{-1} at $[H_2SO_4]= 0.8$ M, $[BrO_3^-]_0 = 93.2$ mM and $[MA]_0 = 46.6$ mM; while the slowest moving center with $< v >= 0.13$ mm min^{-1} is found at almost identical reactant concentrations $[H_2SO_4]= 0.8$ M, $[BrO_3^-]_0 = 93.2$ mM and $[MA]_0 = 93.2$ mM. See the trajectories $ru213_3$ and $ru208_3$ in Table 1. In both experiments the velocities persisted for more than 45 min. The speed of the moving spiral centers can be grouped in three different types, as it is illustrated in Figure 3.8 for three typical systems. The mean speed of the center shown in Figure 3.8 (a) is initially decreasing, later it becomes almost constant. This is the most common development observed. In Figure 3.8 (b) the speed oscillates slightly around its mean value throughout the experiment, and the spiral moves with almost constant velocity. The fluctuations at the end of the detection period are due to noise. The mean speed of the center shown in Figure 3.8 (c) initially is growing until it passes

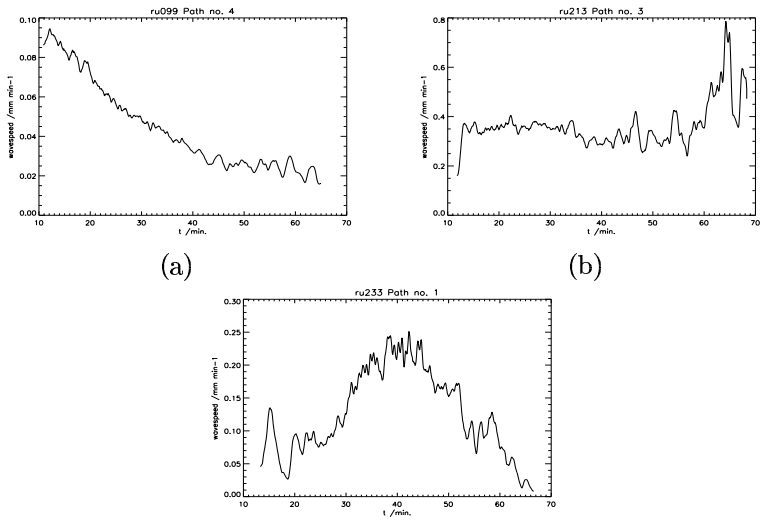


Figure 3.8: The time development of the speed of three spiral centers detected in the experiments ru099, ru213 and ru233 with the following initial conditions. (a): $[H_2SO_4]= 0.4$ M, $[BrO_3^-]_0 = 93.2$ mM, $[MA]_0 = 93.2$ mM. (b): $[H_2SO_4]= 0.8$ M, $[BrO_3^-]_0 = 93.2$ mM, $[MA]_0 = 46.6$ mM. (c): $[H_2SO_4]= 1.0$ M, $[BrO_3^-]_0 = 93.2$ mM, $[MA]_0 = 93.2$ mM.

through a maximum and decreases to the original value again. The trajectory corresponding to this velocity profile is seen in Figure 3.7 (e).

The directions in which the spiral centers move can also be grouped into three types. These are shown in Figure 3.9. The most typical shapes of the trajectories are slightly curved paths as shown in Figure 3.9 (a). This path is 20.6 mm long, and the spiral center is first detected in the interior of the dish. The center is also seen in Figure 3.7 (d). When several centers are detected in the same area of the reaction layer they will most often move in the same direction, as it is seen in the experiment ru101 in Figure 3.7 (d) and (f). The trajectory shown in Figure 3.9 (b) is an example of a more curved path, where the direction of movement turns nearly 360 degrees within the 54 min the center is observed. The characteristics of this center and a nearby detected center are both listed in Table 3.1. These two centers have almost identical mean speeds. In Figure 3.9 (c) an example of a *s*-shaped curve is shown. This spiral center is observed in 1 hour, but the speed of the center is slow.

In several cases, spirals are initially formed as pairs of counter rotating centers. Each spiral can then grow, if the distance between the centers grow during time. We have investigated the double spiral centers formed spontaneously in the experiments in order to detect similarities and differences in such pairs. In Table 3.2 we list the characteristics of 6 centers. The first 4 centers are formed as pairs. The 2 last centers, which develop closed to each other in the experiment ru213, are listed for comparison. In Figure 3.10 the typical development of the trajectories of a double center is shown in (a). In (b) the distance between the 2 centers as function of time is shown. Spontaneously formed double centers are common in our experiments, but many centers do not move away from each other. If the spiral centers, however, drift away from each other the separation distance grows linearly with constant speed as shown in the plot. In Figure 3.10 (c) an example of two atypical centers are shown. These centers are also initially formed as a pair.

The diameter of the petri dish is 9 cm. In most cases the we find the direction and velocity of different centers are related over short

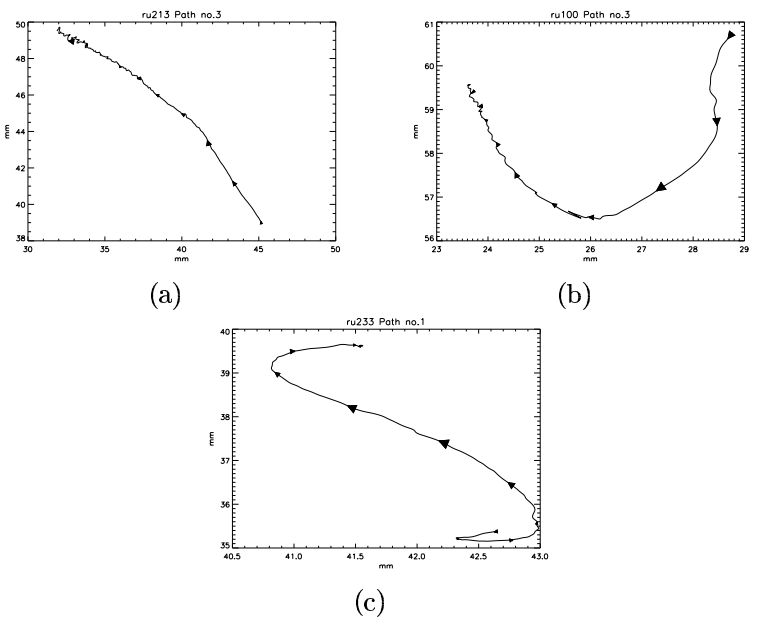


Figure 3.9: Three typical trajectories of spiral centers. The initial conditions are: (a): $[H_2SO_4]$= 0.8 M, $[BrO_3^-]_0$ = 93.2 mM, $[MA]_0$ = 46.6 mM; (b): $[H_2SO_4]$= 0.4 M, $[BrO_3^-]_0$ = 93.2 mM, $[MA]_0$ = 93.2 mM, and to the (c): $[H_2SO_4]$= 1.0 M, $[BrO_3^-]_0$ = 93.2 mM, $[MA]_0$ = 93.2 mM.

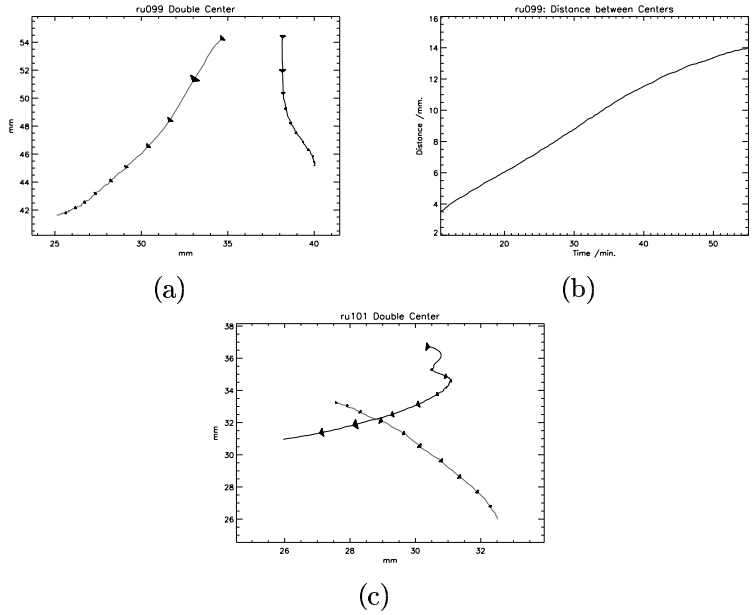


Figure 3.10: In (a) is shown the trajectories of two spiral centers initially formed as a double spiral with reactant concentrations $[H_2SO_4]$= 0.8 M, $[BrO_3^-]_0 = 93.2$ mM, and $[MA]_0 = 93.2$ mM. In (b) the distance between the two centers shown in (a) are calculated as function of time. It is seen that the distance between the centers grows almost linearly. In (c) are the trajectories of a different pair of centers shown. They were initially formed as a double center and move in a fashion so their trajectories cross.

Name:	$[H_2SO_4]$ /M	$[BrO_3^-]_0$ / mM	$[MA]_0$ / mM	Length / mm	Time /sec	$< v >$
$ru099_3$	0.4	93.2	93.2	7.6	33.0	0.23
$ru099_8$	0.4	93.2	93.2	16.6	54.0	0.31
$ru101_3$	0.4	93.2	46.6	9.1	34.0	0.26
$ru101_4$	0.4	93.2	46.6	9.3	54.0	0.17
$ru213_3$	0.8	93.2	46.6	20.6	46.0	0.45
$ru213_4$	0.8	93.2	46.6	12.9	36.0	0.35

Table 3.2: Characteristics of the paths traced by two double centers in the experiments ru099 and ru101 and two single centers in the same part of the dish in the experiment ru213. The speed is of the individual centers. The indices in the first column refer to different paths in the same experiment.

distances, but unrelated over distances comparable to the diameter of the petri dish. It is not likely that convection in the reaction layer causes the movements of the centers, since the reaction layer is very thin. This is in agreement with observations made by e.g. Rodriguez and Vidal (Rodriguez and Vidal, 1989). The zones of coherence which are spontaneously established are separated by zones of annihilation of waves. The coherence zones form a superstructure which persists over long time but change eventually. Such coherence zones can be seen in all experiments shown in Figure 3.7.

3.4 Discussion

A system for automatic tracing of large scale dynamics of spiral and target waves has been presented. It uses a new operational definition of the center of spiral and target waves based on the evolute of the waves. Although computation of the evolute as presented here uses up to fourth order spatial derivatives, it is very stable both with respect to contrast and noise. This is due to the use of the Linear Scale-Space techniques and natural integration over a large support.

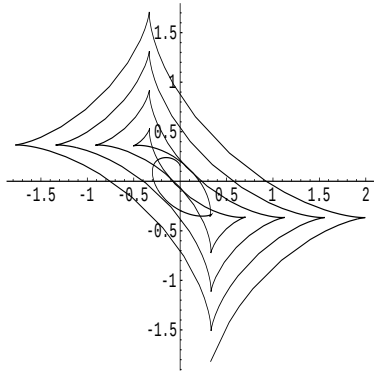


Figure 3.11: The resulting evolute when the aspect ratio of the coordinate system is 1.1, i.e. $f(s) = [s\cos(s), 1.1s\sin(s)]^T$. This evolute has no limiting area.

Note however, the evolute of imbedded ellipses (oval target pattern) ($f(s,k) = [\cos(s), k\sin(s)]^T$, $k \neq 1$) and an 'oval' spiral ($f(s,k) = [s\cos(s), sk\sin(s)]^T, k \neq 1$) does not have compact support with a well defined density maximum. See e.g. Figure 3.11 where an example of the evolute for an oval spiral is shown. This does not seem to be a problem for the real target and spiral waves observed in these experiments, which evolve towards regular patterns with compact support.

In the current implementation the processing time is approximately 40 seconds for a 256x256 image using a Matlab 5.1 implementation running on a HP9000s899 under HPUX 10.20. A preliminary study indicates that it may be possible to reduce the processing time by an order of magnitude.

For future development we note that spiral and target waves can be distinguished by a simple analysis of the neighbourhood of the center: Draw a circle of radius larger than the wave length around the center and count the number of transitions encountered. If an odd number of transitions is found, the pattern is a spiral, otherwise it is a target wave. In this way the method enables quantitative and automatic

identification of spiral and target patterns in experimental data.

By preliminary experiments we expect that other types of geometrical wave configurations may be identified by the use of Linear Scale-Space, e.g. the cusp in the interface between two planar waves. This is left for further development.

From the application of the method to large aspect ratio wave patterns in the ruthenium catalyzed BZ reaction we have found no simple correlations between the patterns of waves of the individual centers and the concentrations of the main reactants. We have, however, found a strong indication of local correlation of the speed and movement of centers arranged in a slowly changing superstructure of regions, but a detailed explanation requires more experiments.

Chapter 4

A Note on Differential Corner Measures[1]

4.1 Introduction

Corner detection plays a central role in many image analysis applications ranging from character recognition to landmark identification. The literature on corner detection roughly divides into two classes. Some use explicit models, see e.g. (Rohr, 1992) for an overview. Others use derivative expressions like the Gaussian curvature, the structure tensor (interest operator, second moment matrix), expressions involving the isophote curvature, and the curvature of Canny edges, see e.g. (Rohr, 1994) for an overview.

One subclass of the latter is corners defined as extremal points of the isophote curvature times the absolute gradient length to some power a:

$$C = |\nabla L|^a \kappa = L_w^{a-1} L_v{}^v \tag{4.1}$$

[1] An earlier version of this work has been published in a conference proceeding (Sporring et al., 1998). The current version is submitted for journal publication as: Jon Sporring, Mads Nielsen, Joachim Weickert, and Ole Fogh Olsen, "A Note on Differential Corner Measures".

where we have used notation w being the gradient direction and v the (perpendicular) tangent direction of the isophote in a right hand coordinate system (w, v). Kitchen and Rosenfeld (Kitchen and Rosenfeld, 1982) suggested to use $a = 1$, Zuniga and Haralick (Zuniga and Haralick, 1983) proposed $a = 0$, and Blom (Blom, 1992) and Lindeberg (Lindeberg, 1994) investigated $a = 3$.

The advantage of using a corner measure with $a > 0$ is that the product will focus on high isophote curvatures close to high contrast edges. There are two special values of a that deserve a note: $a = 0$ is invariant under monotonic transformation of the image intensities (morphological invariance), and $a = 3$ is invariant under affine transformations (the angle of the corner).

We will investigate the above subclass of corner measures in an embedded Gaussian Scale-Space (see (Weickert et al., 1997a) and the references therein):

$$L(x, t) = G(x, t) * L(x)$$

where the original image $L(x)$ is convolved with a Gaussian G of variance $2t$.

The advantage of such an embedding is that it reduces the grid and noise effects and allows for a uniform analysis of corners of all sizes or resolutions. The disadvantage is that the corners are dislocated at high scale and should be traced back to low scale in order to improve their location. We will show that this process – although common in the literature – is problematic due to the complicated catastrophe structure across scale.

We will sketch the catastrophe structure in two different settings. Firstly, by examining the spatial singularity structure of the corner measures. However, Rieger (Rieger, 1992) noted that such corner points usually do not correspond to corners of Canny edges. Therefore in a second approach we will extend Rieger's analysis of corners on Canny edges to edges defined as single isophotes.

Related to this work in terms of Catastrophe Theory is Damon (Damon, 1997), Rieger (Rieger, 1992; Rieger, 1995), Griffin & Colchester (Griffin and Colchester, 1995), Olsen (Olsen, 1997), and Johansen (Johansen, 1997).

4.2 Image structure

Assume a multi-scale 2D image $L(x, y, t) : \mathbb{R}^2 \times \mathbb{R}_+ \mapsto \mathbb{R}$, where x, y are the spatial coordinates and t a scale parameter. We are interested in spatial point features defined as the intersection of zero loci of two differential expressions: $A(x, y, t) = 0$ and $B(x, y, t) = 0$. In our case of corners, this may be $A = \partial_x C$ and $B = \partial_y C$, where $C = |\nabla L|^a \kappa$, $a \in [0; 3]$, or in the case of corners constrained to a single isophote: $A = L - I_0$ and $B = \partial_v C$. We analyse the scale-space curves satisfying $A = B = 0$. In the scale-space points where the tangent of the curve is not pointing directly in the scale direction, we may introduce a local parametrisation of the curve

$$A(x, y(x), t(x)) = 0,\ B(x, y(x), t(x)) = 0$$

such that it is identified by the two scalar functions $y(x)$ and $t(x)$. By differentiation with respect to x and solving a linear system of equations, we obtain:

$$t_x = \frac{A_x B_y - B_x A_y}{A_t B_y - A_y B_t}$$

The denominator is only zero when the tangent of the curve points in the scale direction, and we find that the curve is horizontal only when the numerator is zero. These horizontal points corresponds generically to two curves that meet at one scale yielding an annihilation or creation of a pair of feature points. Whether it is an annihilation or creation for increasing scale can be accessed through the sign of t_{xx}: negative for annihilation and positive for creation. If the second order structure t_{xx} vanishes, we get an event of even higher order.

The Gaussian scale space image satisfies the heat equation ($\partial_t = \sum_i \partial_{x_i x_i}$), changing the general program of catastrophe theory slightly (Damon, 1997). To describe the local jet in space and scale we develop the image in heat polynomials, i.e. polynomials satisfying the heat equation. In 1D they can be generated by the following recursion formula

$$v_n = x v_{n-1} + 2(n-1) t v_{n-2}, \quad v_0 = 1, \quad v_1 = x.$$

A local polynomial model of the 2D image may then be constructed from the jet (the local derivative structure) as

$$\tilde{L} = a_{00} v_0 + a_{10} v_1(x) + a_{01} v_1(y) + a_{20} v_2(x) + a_{11} v_1(x) v_1(y) + \ldots,$$

where the a's are constants proportional to the spatial derivatives in $(x, y, t) = (0, 0, 0)$ by factorial factors.

Given $\tilde{L}$, we compute $\tilde{A}, \tilde{B}$. We count the linear constraints on the jet (the a's) to be satisfied for a given event to happen. In general we have 3 translational degrees of freedom of the coordinate system so that in a generic image we can satisfy linear constraints on three different coefficients. Furthermore we can choose the spatial rotation of our coordinate system freely to simplify expressions.

We have analysed the corner definitions outlined above. In all cases, the curves have generically points at which the curve's tangent has no scale component and second order curve structure such that *both creation and annihilation will genericaly happen*. On top of this, the case $a = 3$ has higher order events happening in critical points of the image, but they are of no interest when considering maxima of $|C|$ since $C = 0$ here. The events for critical points in C are the approach of a saddle to a minimum or maximum. The events for the isophote constrained measure is that a minimum on the curve meets a maximum. In both cases, there is no constraint on whether minima must be positive or maxima negative or vice versa. Hence we can conclude that both annihilation and creation happens generically involving maxima of the absolute value of C. The implication of this is that *tracking a corner over scale can only be performed over an open interval of scales*. At scales outside this interval, the corner does not exist. In Figure 4.1–4.3, we have shown the critical points of C on the letter 'c' over scale for the four different corner measures, and in Figure 4.4–4.5 we show a zoom of a particular interesting set of critical points.

4.3 Experiments on Characters

The experiments we have performed are on binary images of characters. The quantization implies that the images are non-generic at lowest scale but will behave as a generic image ($L_w = 0$) as the scale is increased. This further implies that edges ($L_w = 0$), which initially will be close to the mid-isophote (midway between light and dark), and we may hence approximate their behaviour as the behaviour of the mid-isophote. This also suggests that the image will evolve initially according to Eucledian Shortening Flow, since the isophotes evolve according to $\partial_t S = \left(\kappa + \frac{L_{ww}}{L_w}\right)\vec{N}$ in Linear Scale-Space, where S is an isophote, κ is its curvature, and $\vec{N}$ is its normal (Osher and Sethian, 1988). We will thus expect creations to be high scale phenomena.

In Figure 4.1–4.6 some experiments on the letter 'c' are shown. From these experiments we conclude that both the spatial extremal and the single isophote approaches display similar behaviour w.r.t. the following points: Creation events occur, localisation is poor at high scale, and finally, the number and localisation of critical points at low scale is similar for all a, but the evolution is very different.

Conversely, the spatial extremal approach is very sensitive to noise with respect to the topology (notice the shear explosion in critical points in Figure 4.1 in comparison with Figure 4.2). The single isophote approach is so stable that we have chosen not to display the noisy version of Figure 4.5; there was no visual difference.

Finally, we conclude by Figure 4.6 that when the corners of a single isophote are ordered according to their absolute strength, varying a changes this ordering. Notice especially that the peak at approx. arc-length 40 is practically removed when a is increased while its neighbour at approx. arc-length 20 becomes the dominating corner point for $a = 3$.

4.4 Summary

We have studied the family of corner measures $\kappa|\nabla L|^a$ for $a \in \{0, ..., 3\}$ embedded in Gaussian Scale-Space. Two approaches have been used:

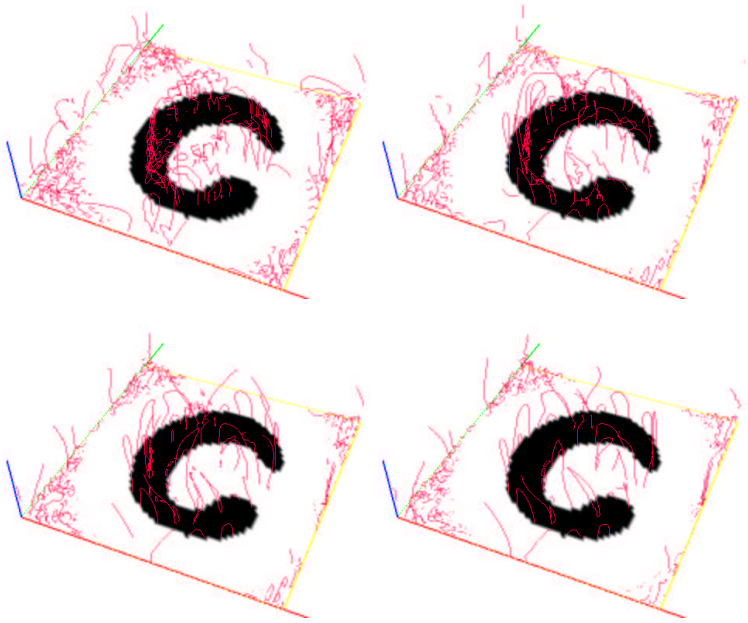


Figure 4.1: Critical points of the noiseless image of 'C' for $a = 0, \ldots, 3$ counting from right to left and top to bottom.

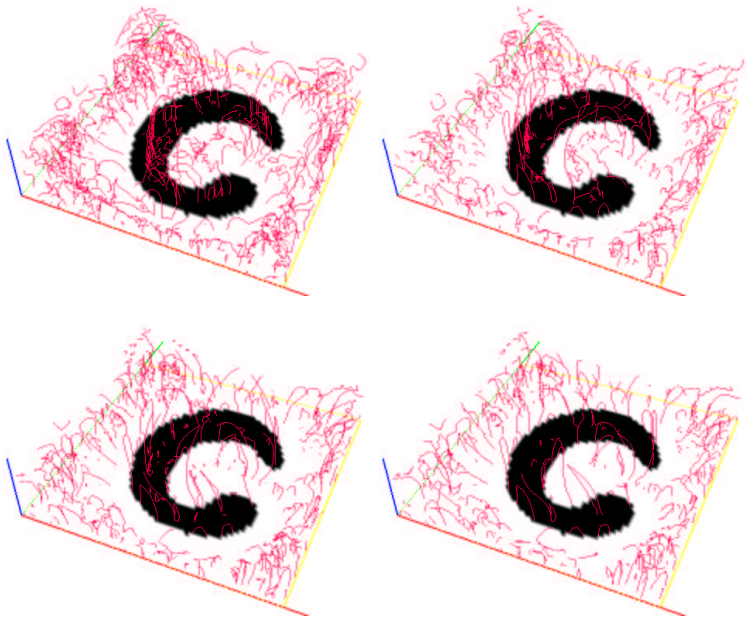


Figure 4.2: Critical points of a noisy image of 'C' for $a = 0, \ldots, 3$. The noise is identically and independently normal distributed noise with mean 0 and standard deviation 5.

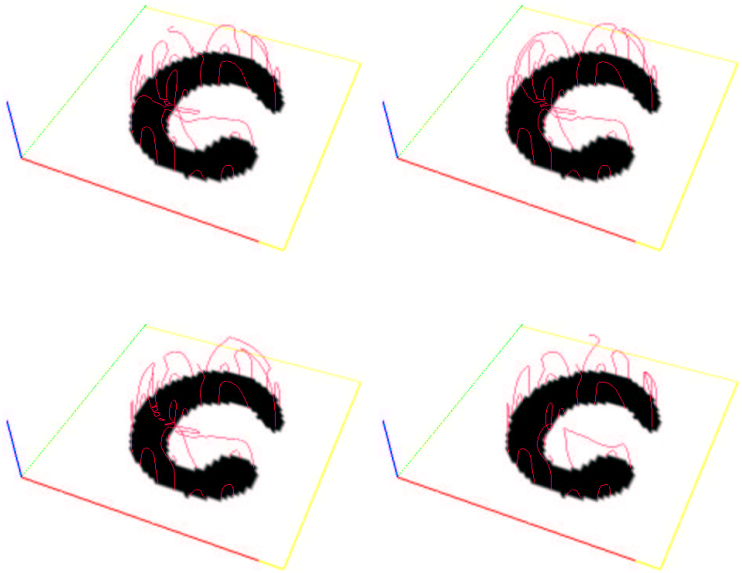


Figure 4.3: Critical points of the mid-isophote of 'C' for $a = 0, \ldots, 3$. The structure for the noiseless and noisy image are visually equal.

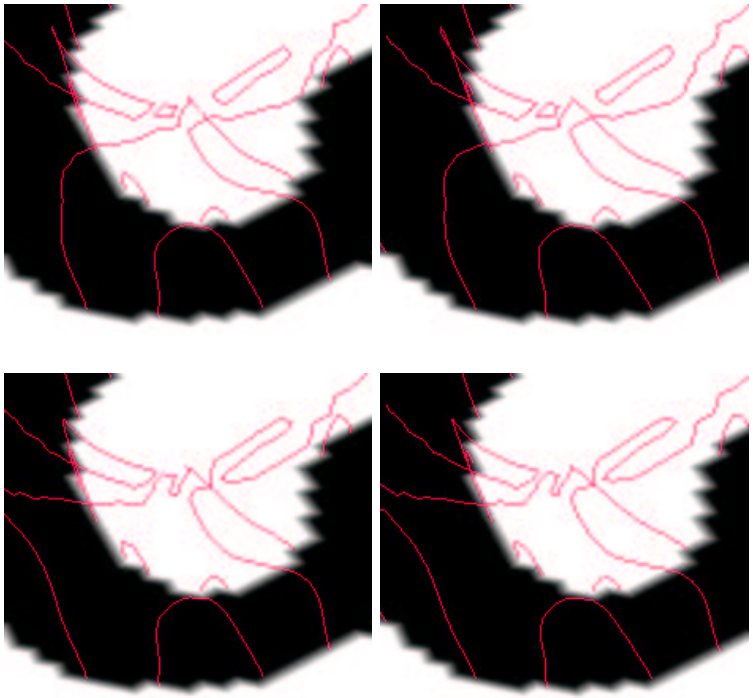


Figure 4.4: Stereo pairs showing a zoom of Figure 4.1 for (TOP) $a = 0$ and (BOTTOM) $a = 1$.

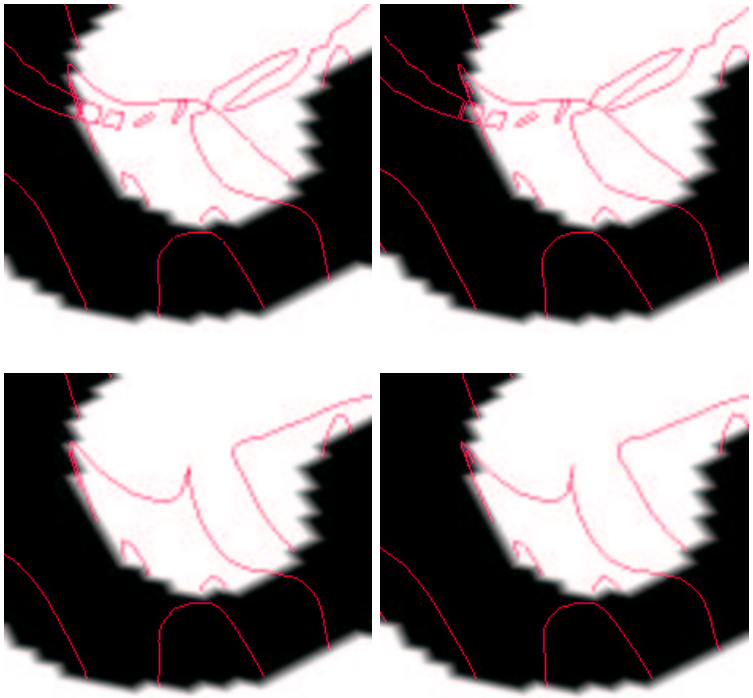


Figure 4.5: Stereo pairs showing a zoom of Figure 4.1 for (TOP) $a = 2$ and (BOTTOM) $a = 3$.

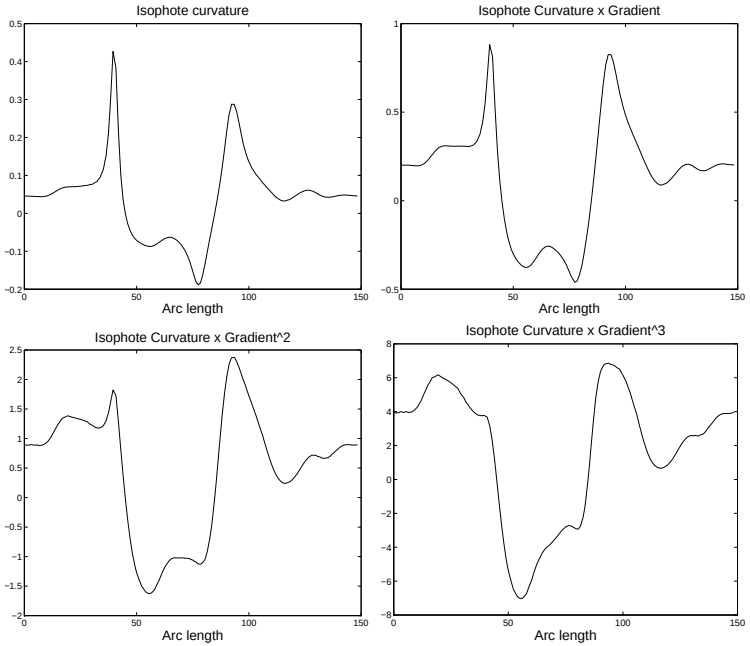


Figure 4.6: $L_w^{a-1}L_{vv}$ for the letter c at $t = 61.8$ and for $a = 0, \ldots, 3$. For the isophote curvature ($a = 0$) the arc-length functions begins on the outer side of 'c', reaches the sharpest corner corresponding to the maximal peak, travels along the inner side of 'c' yielding negative curvature and reaches the onset of the outer part again at approximately arc-length 90. The same arc-length function is used for the other corner measures as well.

The catastrophe structure of both the spatial extrema of the corner measure and the extrema along an isophote.

For both approaches we have concluded that the value $a = 1$ seems to be the simplest with respect to the number of catastrophes. Especially for $a = 0$ many catastrophes appear.

Finally, the isophote approach has been shown to shift the focus away from high isophote curvature points for $a > 1$. The resulting corners do not correspond well with intuition.

Chapter 5

A Piecewise Polynomial Blob Representation[1]

5.1 Coding Office Documents

Many office documents scanned for electronic storage or transmission consist mainly of black and white text and figures. Such data are often the result of a geometrical description. For instance, characters in the Postscript language and the MetaFont program are represented by a collection of polynomials, and figures often consist of line drawings.

A full page is roughly ninety square inches, and with a scanning resolution of six hundred dots per inch this corresponds to thirty million black and white pixels or four megabytes. On the other hand, a full page of text consists of only approximately five thousand characters out of an alphabet of about two hundred and fifty possible. That is, five kilobytes of information.

It would seem that storing or transmitting a document as black and white pixels is grossly wasteful. But, the page carries other information than just the characters. It is set in a certain font and with a certain

[1] An earlier version of this chapter has been published as a technical report (Sporring, 1998).

page design, which if neglected seriously worsen reading quality.

Several compression systems have been suggested to date. Some systems try to identify the font and then encode the document as ASCII augmented with character placement, but since the number of fonts increase each day, the task becomes increasingly difficult. Also, errors in these systems, where the wrong font or type is identified, are very disturbing to the human eye.

The most popular systems are based on the algorithm CONTEXT (Rissanen, 1983). This algorithm completely disregards the geometrical content and compresses solely based on the statistics of the neighbouring pixels. Such systems are highly successful in terms of compression, but they have some disadvantages. Firstly, it is a one dimensional system which scans the document line wise, partitioning the document into what has been read and what has not. I.e. the full two dimensional structure is not used. Secondly, these systems do not use a model class which is close to what originally produced the data. At low scanning resolution, the discretization noise will be dominating which corresponds well with the model class used by the algorithm CONTEXT, but at high scanning resolution the noise will be less prominent. Hence, it is possible to do better by choosing a geometrically based model class. Finally, the model is not present in an analytical form. Therefore, there is only a limited possibility to decode at various resolutions.

This report is on the compression of blobs or more precisely of coherent structures such as a character. The blobs are representable by their contours, since each contour is closed, and the filling of the space between contours can easily be asserted, e.g. by the odd/even fill rule: Examine any straight line on a page starting from a known colour, and flip colour each time a contour is crossed.

The blobs are segmented and combined into an alphabet of symbols by an external process. The goal of this work is to code the alphabet in a lossless manner, by splitting the code into an analytical model and a noise signal. It is the underlying intent to investigate the usability of the model alone as a lossy code.

The list of literature, where geometrical descriptors are studied, is extremely long, but we especially found (Lindeberg and Li, 1997; Rosin

and West, 1995; Chen and Chin, 1993) to be valuable in this context. It seems that the novelty of this work is to combine the myopic view of differential geometry with information theory to gain a connection between local and global models. Connecting local and global is not possible without comparison of description complexity with the error of the approximation. The Minimum Description Length methodology (Rissanen, 1989; Rissanen, 1996) may be the best for this means.

This chapter is organised as follows. First we will introduce the imaging model of Linear Scale-Space, and demonstrate how this model enables us to extract geometrical information from the image in a consistent fashion. Then we will discuss various geometrical descriptors in relation to compression and finally we will discuss the theory of descriptive complexity and MDL and we will demonstrate an algorithm for compressing images of text on an alphabet of 158 blobs of various size.

5.2 Linear Scale-Space Analysis

The shapes to be coded will be characters formed by a collection of polynomials, converted to very high resolution raster and printed, and finally scanned by a fax machine at high resolution. The result is a binary function sampled on discrete grid also called the image. To ease geometrical measurements on such data we will use the theory of sampling called Linear Scale-Space.

Linear Scale-Space was first introduced by Iijima in 1962 (Iijima, 1962; Iijima, 1971; Iijima, 1972; Weickert et al., 1997a) as a descriptive tool for signal analysis, and then rediscovered in 1983 by Witkin and Koenderink (Witkin, 1983; Koenderink, 1984). Linear Scale-Space is an axiomatic derivation of an imaging model. The model is useful in several ways. Firstly, it greatly reduces grid effects, which is a direct result of the sampling process and hence has only little to do with the process being sampled. Secondly, it allows for a well-posed differentiation of sampled signals, which in turn allows for the use of differential geometrical tools on sampled signals. Finally, it is an algorithmically unifying approach to analyse signals and images for

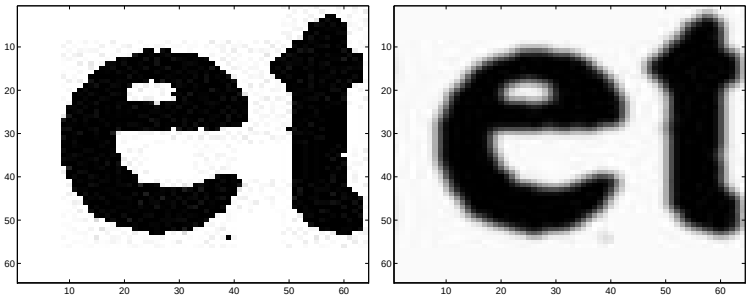


Figure 5.1: The original and smoothed image at scale 1/2

contents of different sizes (Florack, 1997).

We will now shortly review the basis of Linear Scale-Space. The essence of Linear Scale-Space is smoothing with isotropic Gaussian kernels,

$$L = I * G(t), \tag{5.1}$$

where $*$ is the convolution operator, and

$$G(t) = \frac{1}{(\pi 4t)^{D/2}} e^{\frac{-||x||_2^2}{4t}}, \tag{5.2}$$

where D is the dimensionality (2 in the present case), and $||\cdot||_2$ is the Euclidean length operator. The standard deviation is $\sigma = \sqrt{2t}$. The effect for $t = 1/2$ is shown in Figure 5.1. The Gaussian kernel is the Green's function of the Heat Diffusion Equation,

$$\partial_t = \sum_i \partial_{x_i x_i}, \tag{5.3}$$

where the right-hand side is understood to be the sum over all dimensions at hand, e.g. $\partial_{xx} + \partial_{yy}$ in the present case. This diffusion equation is particularly sensible to study because of the following properties (Koenderink, 1984):

- Invariance to translation and rotation.

- Causality (in 1D this is equal to the non-increasing of critical points, and in ND this translates into: all isophotes are upward (large t) closed).
- Treats all scales (t) equally.

We here treat the initial image as the boundary point of the diffusion equation, i.e. $L \to I$ for $t \to 0$. The uniqueness of the Gaussian kernel in the linear setting can also be derived from other sets of axioms, see (Weickert et al., 1997a) for a review.

Since Linear Scale-Space uses the Gaussian kernel and this kernel obviously has exponential decay, Linear Scale-Space is a very convenient tool for studying differential geometry of data of polynomial order. That is by partial differentiation it is seen that

$$DL = D(I * G) = I * DG, \tag{5.4}$$

where I is the sampled signal or image and D is any linear differential operator. In contrast to many other methods of calculating derivatives this one is well-posed in the sense of Hadamard (Hadamard, 1902).

5.2.1 Differential Geometry on Discrete Data

To illustrate one of the above points, we will now review a number of non-linear differential operators used to extract geometrical structure in 2D scalar image data.

A commonly used edge operator is the maximum of the absolute gradient magnitude,

$$G = \sqrt{L_x^2 + L_y^2}, \tag{5.5}$$

where L_{x_i} is the derivative of the image in the x_i'th direction. For the present example the absolute gradient magnitude is shown in Figure 5.2.

Another operator is the isophote-curvature which conveniently can be calculated as,

$$\kappa = \frac{2L_xL_yL_{xy} - L_{yy}L_x^2 - L_{xx}L_y^2}{(L_x^2 + L_y^2)^{3/2}}. \tag{5.6}$$

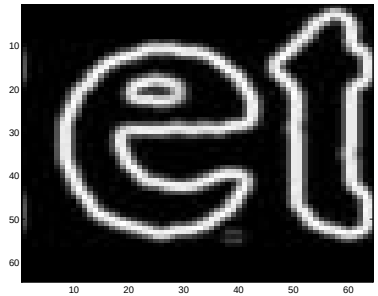


Figure 5.2: The gradient magnitude at scale 1/2

The isophote-curvature is undefined in critical points where the gradient magnitude is 0, and it is numerically unstable for small gradient magnitudes. To ensure that corners are placed at high gradient magnitudes similar corner detectors have been suggested, which multiply κ with the gradient magnitude to some power, i.e. $\kappa_a = \kappa G^a$, $a \in \{1, 2, 3\}$. Unfortunately, these rearrange the ordering of the corners and it seems that the process of tracing corners from high to low scale can become very difficult (Sporring et al., 1998). Figure 5.3 shows the isophote-curvature at reasonably large gradient magnitudes. For this particular image the selected image points have an accumulated frequency of absolute isophote-curvature values as shown in Figure 5.4. Noted should be that the accumulated frequency curve is almost flat for absolute curvature values above 0.5. Although Gaussian smoothing is not very good for detecting corners, this distribution is an indication of the relatively few corner marks in the image detected by κ.

5.2.2 Noise and Derivatives

Although the Gaussian function generates high frequency suppressing derivatives, there are several factors to consider. One is the aliasing error due to a necessary assumption of band-limitation. In the Frequency domain, differentiation can be seen to equal a multiplication with the function $(\omega i)^n$, where ω is the angular frequency, $i = \sqrt{-1}$,

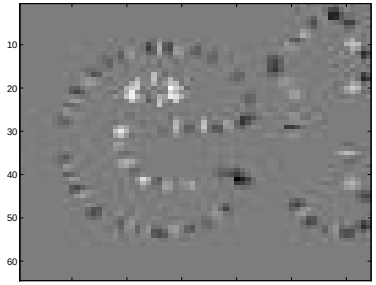


Figure 5.3: The isophote-curvature at reasonably large gradient magnitudes at scale 1/2. White corresponds to positive curvatures, while black points are negative curvatures with respect to the isophote normal vector.

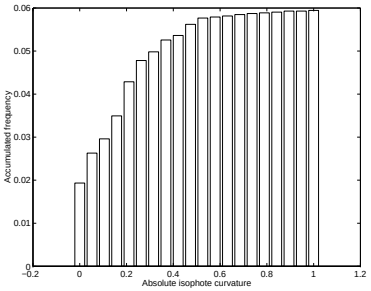


Figure 5.4: The accumulated frequency of the isophote-curvatures at reasonably large gradient magnitudes at scale 1/2.

and n the differentiation order. This effect has been studied by Haar Romeny *et al.* (Haar Romeny et al., 1994) and they express the aliasing error as the power of the aliased frequencies relative to the total power is given as,

$$\text{error}(n,t) = \frac{\int_{\pi}^{\infty} \omega^{2n} e^{-t\omega^2/2}\, d\omega}{\int_{0}^{\infty} \omega^{2n} e^{-t\omega^2/2}\, d\omega}, \tag{5.7}$$

where n is the differentiation order, t the scale, and ω the angular frequency. This error is generally relatively unimportant even for very high derivatives at a comparatively small scale, e.g. the relative error of a 100 fold derivative at scale 6.5 (standard deviation of 3.6) is less than 1%.

The second type of errors is understood through the additive noise model. Here noise is seen to be a (usually) low amplitude but high frequency signal added to the original signal. Again due to the polynomial behaviour of the differentiation operator in the frequency domain, this noise will be amplified as the differentiation order increases. Blom *et al.* (Blom et al., 1993) have studied the propagation of noise in relation to the spatial derivatives in Linear Scale-Space in terms of the momentum of the noise of derivation. The results are quite complicated, and we will here just summarise the simplest case of independently identically distributed noise,

$$M^2_{n_x,n_y} = \epsilon^2 \langle N^2 \rangle Q_{2n_x} Q_{2n_y} \frac{}{2\pi t^{1+n_x+n_y}}, \tag{5.8}$$

where n_x and n_y are the differentiation orders, ϵ is the distance between samples, $\langle N^2 \rangle$ is the mean of the squared noise, and Q_n is defined as,

$$Q_n = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{if } n \text{ odd} \\ \prod_{i=1}^{n/2}(2i-1) & \text{if } n \text{ even} \end{cases}. \tag{5.9}$$

This of course assumes that the noise characteristics are known (which is never the case except for artificial data) or at least estimated.

5.2.3 From Black/White to Gray and Back

Due to the nature of Linear Scale-Space black/white images are instantly transformed into continuous valued images and although the image is simplified it is not clear how to make use of this simplification. In contrast, the relation to the original black/white images is simpler for morphological scale-spaces such as suggested by Boomgaard *et al.* (Boomgaard et al., 1996). We have chosen not to utilise these scale-spaces in order to be able to emulate a sampling at different resolutions, i.e. to take advantage of the implicit analysis of the catastrophe structure imposed by sampling, see e.g. Damon (Damon, 1997) for more details. Hence we will assume that the sampling device has a global threshold and we therefore need not consider edges other than isophotes. We will in the following investigate a binarization of the continuous valued images.

Viewing the image as landscapes with the intensity as the height function, a slice of an ideal edge that is one with infinite extend, would look like the Heaviside function,

$$H(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{otherwise} \end{cases}, \tag{5.10}$$

and the family of functions generated by Linear Scale-Space $H * G_\sigma$ would all have identical values $\int_0^\infty G_\sigma \, dx = 1/2$ in 0. This would lead to the conclusion that the 1/2 isophote (equal intensity value) should be used to distinguish black from white. This is shown in Figure 5.5. But black and white images are not made from a collection of ideal edges, rather from box-like functions,

$$B_w(x) = \begin{cases} 1 & |x| < w/2 \\ 0 & \text{otherwise} \end{cases}, \tag{5.11}$$

This family of functions is significantly different for σ approximately equal to or larger than w in that it rapidly approaches a Gaussian distribution and hence the values of $B_w * G_\sigma \to 0$ for $\sigma \to \infty$. Hence the 1/2-isophote will have the effect that small blobs in comparison to the scale will be ignored or removed. A second possibility is to choose the threshold to be midway between maximum and minimum,

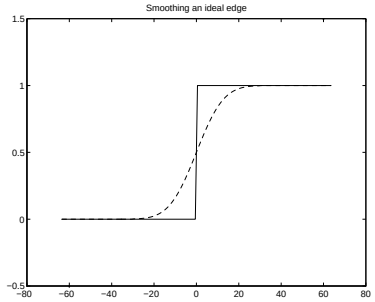


Figure 5.5: The Heaviside function and a smoothed (dashed) version

but the effect of this choice is difficult to predict for images consisting of more than one blob. Finally, a third choice might be to use the mean value as threshold value, since the ideal box model assumes the image to be zero beyond its border implying that the mean value is 0. In the common Fourier Transformation implementation, the image is assumed to be periodic and the mean value is then not zero, i.e. the function converges to the mean value instead of 0. Such a choice would make the result depend on the black to white ratio which is arbitrary for office documents. These images might as well be modelled to have the colour of paper outside the image border. We will in this report use the 1/2-isophote as the representative of a blob on any scale.

5.2.4 Superficial and Deep Structure

Even though Linear Scale-Space simplifies the image contents structure is also dislocated. Thus to focus on large scale structure we may conveniently locate them at high scale, while to localise these same structures in the original image we must track the structure across scales. What can be tracked are points. I.e, on a straight line the only distinguishable points are the endpoints, while most curved lines have singularities in the derivatives that can be tracked. In this work we will focus on extremal points in the curvature function and call them corners.

The study of the family of signals and images generated by the Linear Scale-Space is usually divided into two classes in terms of structure. The superficial structure is the differential structure present at a particular scale, i.e. the configuration of the extremal points, while the deep structure is a study in how the differential structure changes as a function of scale. The last part relies on the theory of catastrophe, where a catastrophe is the annihilation or creation of extremal points. For references see (Gilmore, 1981; Koenderink, 1984; Griffin and Colchester, 1995; Rieger, 1995; Olsen, 1996; Damon, 1997; Sporring et al., 1998).

The catastrophes can be seen to be non-generic in the sense that they exist at each individual scale with probability 0, but since the Linear Scale-Space is a continuous family of functions, the effects of the catastrophes will be felt. Consider the example of a camera filming a piece of wood being broken, and imagine that the process of breaking goes infinitely fast. First the camera will have some images of the wood bending, and at one point the images will show the two pieces, but the actual breaking point will never be caught on film. The breaking event can be considered a catastrophe.

For 1D signals the deep structure in Linear Scale-Space is simple: the only catastrophe that takes place are pair-wise annihilations of a maximum and a minimum, but for 2D images, creations may occur. The annihilations and creations are always occurring in pairs of extrema and saddle points and the catastrophe theory thus suggests a grammar of events. Unfortunately, it is easy to see that at a catastrophe, the involved extrema travel at very large speeds, and the problem of assigning correspondence between scales is generally very difficult to solve (Lindeberg, 1994).

Finally, although a grammar of generic events can be assembled, nothing is told of the close to unlikely events. E.g. it is a non-generic event that all the second order spatial derivatives of an image are zero at some point in the image, but in practice the lines of zero ∂_{xx}, ∂_{xy}, and ∂_{yy} is sometimes seen to pass within a pixel distance of each other, and it even seems that the probability of this happening increases with scale.

We will return to a deeper discussion on these matters in relation to corners later in Section 5.3.3.

5.2.5 Non-linear Scale-Spaces Designed for Curves

While we in this work will emphasise the role of the sampling and hence use the Linear Scale-Space, several non-linear Scale-Spaces have been proposed especially designed for the study of curves. Some authors have investigated the convolution of each coordinate function (Granlund, 1972; Lowe, 1988; Mokhtarian and Mackworth, 1992; Oliensis, 1993), which is not a rotational invariant scale-space, but by constantly resampling of the arch length function and infinitesimal steps it does perform similar to Euclidean Shortening Flow as described below (Mokhtarian and Mackworth, 1992). Some have examined convolution of variants of the curvature function (Horn and Weldon Jr., 1986), and some have chosen to work directly on the curve in the arc length parameter (Alvarez and Morel, 1994; Sapiro and Tannenbaum, 1995; Sethian, 1996). The later methods use the geometrical evolution equation,

$$\partial_t = \beta \partial_s{}^s, \tag{5.12}$$

where t is time and s is arc-length. Usually β is taken to be the isophote curvature, and the Scale-Space is then called Euclidean Shortening Flow. Note especially, β's can be defined to preserve the area of closed curves (Sapiro and Tannenbaum, 1995) under various projections.

Osher and Sethian (Osher and Sethian, 1988; Sethian, 1996) have shown that the geometrical evolution equation on all isophotes in an image can be calculated as,

$$I_t = \beta |\nabla I|, \tag{5.13}$$

where $|\nabla I|$ is the gradient length image. Also, fast algorithms have been devised approximating the geometrical evolution equation for a single isophote in an image by Sethian and others (Sethian, 1996).

These methods will evolve any closed curves smoothly into a circle or a point and are thus simplifiers. But for the purpose of the algorithm to be developed later in this work, this type of smoothing is of less interest. It is the behaviour of shapes under image and signal sampling

that we choose to focus on in order to mimic fax-like processes at different sampling resolutions.

5.3 1+1D and 2D Contour Models

In the previous chapter we described a well-posed model for image formation which has the intrinsic property that differential geometric measurements are also well-posed. We will now discuss various differential geometric methods for two dimensional shape description.

There are basically two ways of viewing curves in two dimensional space. Either as a tuple of coordinate functions $[x(s), y(s)]^T$ where s is an arbitrary step-function or as the curvature function $\kappa(s)$ as function of the arc-length.

The tuple perspective we immediately disregard since it is not rotationally invariant. The analysis and coding we wish to perform should not depend on the rotation of the blob, i.e. the positioning of the office document in the scanner or fax machine.

The Fundamental Theorem of Curves states that any continuous two times differential 2D space curve can be described by the curvature functions up to a Euclidean movement (Koenderink, 1990). In the (local) Frenet coordinate system this implies that the curves are locally well represented by the Frenet approximation,

$$\vec{x}(s) \simeq \vec{x}(s_0) + s\vec{T}(s_0) + \frac{s^2}{2}\kappa(s_0)\vec{N}(s_0), \tag{5.14}$$

where $\vec{T}$ and $\vec{N}$ are the normalized tangent and normal vectors for the curve.

Although the curvature uniquely describes the intrinsic properties of the curve, it might not be the computationally most feasible representation. One major draw-back is that in order to solve the Frenet approximation given a curvature function one has to solve a differential equation and the errors accumulated thereby seem difficult to handle. Further, it is unclear how to incorporate the full knowledge of the limited set of curvature functions that generate closed non-intersecting curves.

An alternative approach is to cut the shape into pieces representable as functions each with its own coordinate system. This is known as a Monge patch. One obvious advantage is that the shape is studied as a one dimensional entity, and although it is not the intrinsic curvature function, each piece will converge towards the Frenet approximation as the density of cutting points (knots) tends to infinity. Further if the knots are distributed according to the absolute curvature, the deviation from the intrinsic shape is greatly reduced. To see this, view the shape in the myopic perspective, i.e. by zooming until everything looks linear. Clearly, a representation by piecewise linear functions differs very little from the intrinsic shape in this perspective. The amount of zooming necessary is a function of curvature: When the curvature is large the zooming has to be great, and vice versa for small curvatures. The major disadvantage is that the cutting is a global process, i.e. each placement of a knot depends on the placement of the neighbouring knots and the curve in between. Secondly, when only few knots are used, the functions between knots are far from the intrinsic shape. E.g. a circle needs at least 2 cuts, while its curvature function is a constant.

To summarise, this approach makes use of a set of coordinate pairs called knots and connecting 1D functions. Please note that although we make use of 2D coordinates this representation is still rotational invariant since the majority of contour points is modelled as rotationally invariant 1D functions, and the knots will be chosen in a rotationally invariant fashion. We call this the 1+1D model.

We will in the following make deeper analysis of the curvature function and the 1+1D model, and we will sketch a contour approximation algorithm.

5.3.1 A Classification of Shape Algorithms

Much effort in the literature on shape approximations has been spent on the study of the contour as the two 1D coordinate functions, but since these functions are not rotationally invariant they are not the intrinsic functions of shape. Whichever functions one chooses to work with the essential structure is of second order. That is to say, in the

following model-review one may think of either each coordinate function in the 2D case, or only the one Monge function in the 1+1D case. In terms of the complexity of the curvature function, a classification of the suggested contour models in the literature is:

- Piecewise zero-value curvature, i.e. polygonal approximation including chain-codes (Freeman, 1961; Ramer, 1972).
- Piecewise constant curvature (different from zero), i.e. circular arcs (Lindeberg and Li, 1997).
- Higher order curvature functions, i.e. polynomials (Chen and Chin, 1993), splines (Boor, 1978), elliptical arcs (Lindeberg and Li, 1997), sinusoidals (Granlund, 1972; Rosin and Venkatesh, 1993), and abstract curvature 'sketch' models (Asada and Brady, 1986; Rosin and West, 1995).

The present work belongs to the last class. The essence being threefold: We will explore the use of Linear Scale-Space to select break points between polynomials, we will use Monge patches as approximations of the curvature to contour problems, and finally we will use descriptive complexity techniques to do model refinement and hence selection. The novelty of the present work is to bridge the gap between the myopic view (local differential geometry) and global models.

5.3.2 The Rod Model

As an illustration let us investigate the simplest case of piecewise linear models: In Figure 5.6 is a piece of the so-called rod model (Koenderink, 1990) shown. One could say that all the curvature information is placed at the joins of the straight lines, and in fact it is sometimes feasible to view the curvature as the limit of α/l for $l \to 0$, where α is the angle between two successive rods and l is the distance between joins.

We can thus represent a contour as successive rod lengths and angle changes. To reconstruct beginning from only one tangent vector as specified by the starting point and the rotation angle, it is necessary to assume the tangent vector at a junction of two lines to be equal to

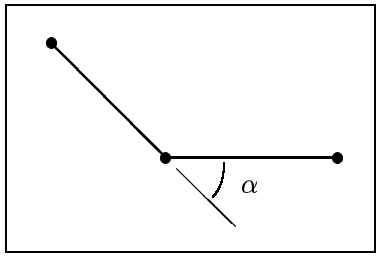


Figure 5.6: The piece-wise linear model showing the first-order effect of curvature as the angular change α.

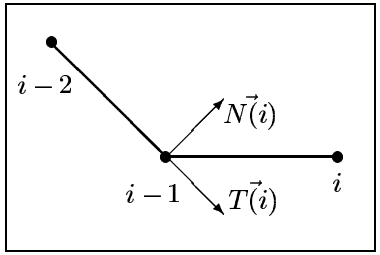


Figure 5.7: The placing of the tangent information in a recursive fashion.

this previous line, see Figure 5.7. Hence a new point can be calculated recursively as,

$$\vec{x}(i) = \vec{x}(i-1) + s(i) \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \cdot \vec{T}(i), \tag{5.15}$$

using,

$$\vec{T}(i) = \frac{\vec{x}(i-2) - \vec{x}(i-1)}{||\vec{x}(i-2) - \vec{x}(i-1)||_2}, \tag{5.16}$$

where $||\cdot||_2$ is the Euclidean length operator.

The reader should note that although in the limit of infinitely small rod lengths, the Frenet approximation and the above sketched algorithm coincide, the Frenet approximation cannot be used as a reconstruction algorithm unless length and angle functions are artificially

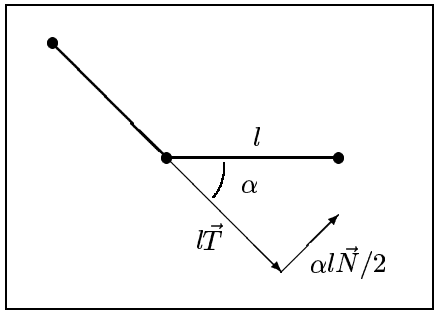


Figure 5.8: The truncation in the Frenet Approximation results in reconstruction errors.

modified to account for higher order structure. The difference is illustrated in Figure 5.8. The error is close to linear in angle and length.

As discussed previously, the essential information of the 2D shape is in the local curvature function. While the algorithm sketched in Equations 5.15 and 5.16 does produce a one-to-one transformation, it is poorly suited for shape analysis. The approximation of curvature as angular changes does not separate the noise due to the discretization grid and that of the original shape. In Figure 5.9 is the approximated curvature plotted as a function of accumulative rod length. The 'true' signal is forever lost at the point of discretization, but the grid effect can be reduced by smoothing, i.e. sampling the same contour at different scales in the Linear Scale-Space, and calculating the angular changes at each scale does have a regularising effect.

The large scale curvature is related to the rim of the letter 'e' as follows: Starting from the left most part of the letter 'e' in Figure 5.1 and going downwards, the first major minimum in the curvature function is at the point where the isophote turns left at the bottom right. Then follows a maximum corresponding to the next right turn, and finally the last significant minimum is the following left turn.

Another common 'smoothing' algorithm is to calculate a local approximation of the contour and evaluate the curvature of the approximation. As an example we have tested a pair of second order polyno-

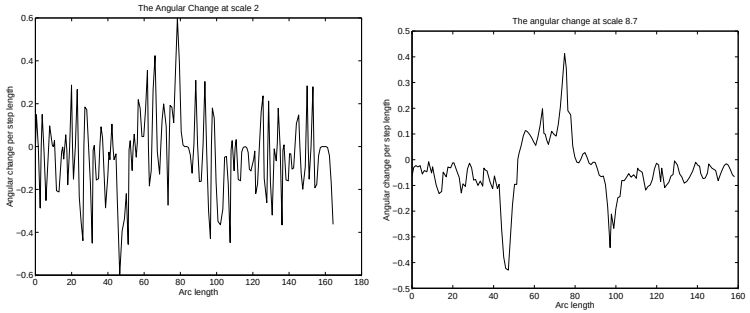


Figure 5.9: The curvature function of the outer rim of the letter 'e' in the example. The curvature is approximated in the rod-model as α/l, and the image is taken at scale 2 and 8.7.

mials, one for each coordinate function, approximated all consecutive sets of 5 points, and fitted the polynomials using the method of linear least square. The curvature can be found as (Mokhtarian and Mackworth, 1992)

$$\kappa = \frac{x'y'' - y'x''}{\overline{(x'^2 + y'^2)^{3/2}}}, \tag{5.17}$$

where x and y are the coordinate functions and the mark denotes their respectively derivatives. The result of this method can be seen in Figure 5.10.

Finally, the Linear Scale-Space imaging model allows for the design of a particular imaging device for measuring the curvature (given in Equation 5.6). The same experiment as above yields curvature functions as shown in Figure 5.11.

Comparing Figure 5.9–5.11 we see that both the polynomial coordinate approximation and the Linear Scale-Space have superior regularization effect compared to the rod model. The polynomial coordinate approximation two disadvantage in comparison with the Linear Scale-Space. Firstly, the coordinate functions are not invariant under rotation of contour hence neither are the polynomial approximations.

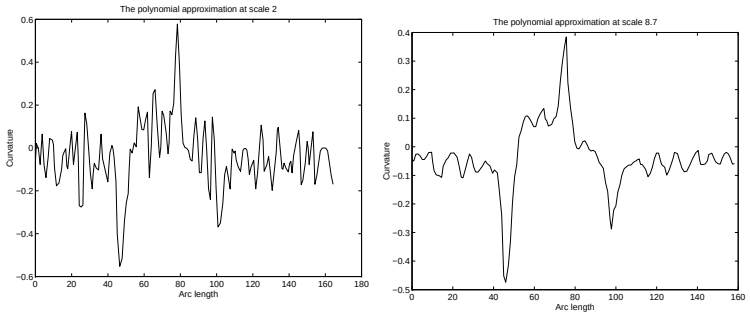


Figure 5.10: The polynomial approximation for estimating the curvature and the effect of Linear Scale-Space.

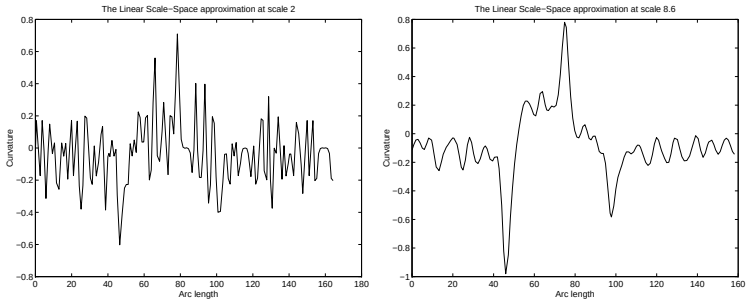


Figure 5.11: The Linear Scale-Space approximation for estimating the curvature at different scales.

Secondly, the polynomial approximation have two unknow parameters: the polynomial degree and the number of points on the contour to include. Based on the above, it is our opinion that Linear Scale-Space is the better of the three for measuring curvature.

5.3.3 A Coarse to Fine Analysis

In order to separate the corners that survive at high scale from those that probably are due to discretization noise, we have to track corners from high to low scale. For this we need a grammar of possible changes in the structure of the corners, i.e. what kind of events can occur during tracking from high to low: Can a corner just disappear? Can two corners annihilate each other? etc.. This is the subject of deep structure.

Although the deep structure of signals and images is fairly well understood, applying this to the catastrophes of corners on an isophote is rather complicated. As an aside, the scale-spaces designed for curves are well understood in terms of deep structure of the curvature function, since its simple 1 dimensional form applies directly. Alas this is not the Scale-Space we have chosen to work with.

There are two levels of deep structure an algorithm has to accommodate. Firstly, the number of isophotes might change, for instance, the hole in the letter 'e' might disappear. This is called a topological change. Secondly, the extrema structure in the isophote curvature can change (Sporring et al., 1998).

The following events can occur with regards to the number of isophotes as the scale increases:

1. Nothing

2. An isophote disappears, which is a very common low scale phenomenon. E.g. a high spike like salt/pepper noise will at some intensity value have a small isophote that disappears quickly.

3. Two isophotes join, this is more common at high scale, where two nearby blobs melts into one below certain intensity values.

4. An isophote splits into two. This is most common when connected blobs of alternating big and small size are present. The classical example is the image of two circles connected by a thin ramp. The ramp is then eroded faster than the circles creating two separate isophotes.

The events above are well understood, and while they can occur at any intensity value, they certainly will affect the chosen isophote. The catastrophe structure of the curvature function for a single isophote is given as follows (Sporring et al., 1998):

1. Nothing
2. A maximum and a minimum pair is annihilated.
3. A maximum and a minimum pair is created.

In terms of analysis, the extremal curvature points both mutate according to the catastrophe structure and move their absolute position. Given a curvature function, a desired result of an analysis is a classification of the extremal points according to their stability with respect to scale changes. I.e. at low scale the positional precision is high, but also the number of extremal points is high. To distinguish we would like to track the extremal points at high scale to low scale, and as such obtaining a ranking of the extremal points at high positional precision.

5.3.4 A Shape Approximation Algorithm

In order to approximate a contour by the 1+1D model, we need to identify a number of knots on the contour and model the contour in between knots by 1D functions. Above is described a method for identifying semantically important points on a contour, and we believe that the placement of these high scale corners are psychophysically more important than the exact approximation of the contour in between, hence we will use the set of high scale corners as knots. But the corner set does not guarantee that the contour in between can be viewed as a 1D function, and we are forced to introduce extra knots.

Several methods have been suggested in the Spline literature, and we choose to sample the integral of square root of the absolute curvature linearly in between knots as suggested by de Boor (Boor, 1978). This implies that extra knots are introduced when there is much curvature which conforms with the myopic view as described earlier.

The algorithm so far is as follows:

1. Calculate a range of image scales and the isophote contour
2. Extract the mid contours and their curvatures
3. Find and classify/track the scale range of each extremum
4. Represent the curves between knots as a 1D function, adding extra knots where either the 1D function assumption is violated or the error is great, e.g. by equal increment in the integral of the square root of the absolute curvature.

The algorithm has 4 parameters: 3 for controlling the sampling in scales and 1 for controlling the frequency of non-extremal knots.

Most importantly is the setting of the sampling range and density in the scale variable. These variables can fairly easily be set a priori depending on which range of blobs one wishes to cover and the original sampling density of the images compared with the sampling density of the originals. I.e. an exponential scale of standard variances $(0.8, 6)$ sampled 3 times is a fair choice. If the scanning has been performed at 600 dpi (assumed to be close to the original printing density), then this will give a sampling in the range (375dpi, 50dpi). The sampling density should be set as close as allowable with respect to computational time, since this effects the precision of the tracking algorithm. In our experiment 3 samples appear to be enough, though.

Finally, the Monge Patch approximation restricts the allowable constant increment in the integral of the square root of the absolute curvature. The smaller this value is, the better the approximation. It is clear that the integral steps cannot be larger than π, and a reasonable guess is to set it to $\pi/2$, i.e. to disallow turns larger than 90 degrees.

5.4 Model Selection by Descriptive Complexity

So far we have discussed various alternative representations and some of the choices made have been hinted upon to have been made from a compression viewpoint. We will now fully illustrate why we believe this to be a very important viewpoint, and orchestrate our algorithm with a fine tuning with respect to compression.

In the following we will use the concept of a coder–decoder pair, the main issue being a reformulation of data into a form that can be reversed to yield the same data again. This is also called lossless coding. This is an important perspective since it highlights the interplay between syntax and semantics.

5.4.1 Kolmogorov and Stochastic Complexity

In the general case, the reformulation is in terms of a particular universal machine, e.g. Turing machines, which can be thought of as a computer language like Pascal, C, etc. The choice of the machine dictates the syntax while the program or reformulation chosen is a semantic choice. There are of course a huge number of programs to express a single set of data.

Independently did Chaitin, Solomonoff, and Kolmogorov discover the concept of Kolmogorov Complexity (Solomonoff, 1964; Kolmogorv, 1965; Chaitin, 1966). It is simply the length of the shortest program that produce a specific set of data and halts. The halting concept reduces the running time to finite time, but it also implies that the Kolmogorov Complexity is non-computable since the problem of deciding if a particular program will halt is generally non-computable. Thus the Kolmogorov Complexity should be seen as a lower unattainable bound for compression.

An important concept in descriptive complexity is the Minimum Description Length (MDL) principle (Rissanen, 1983; Rissanen, 1989) also called Stochastic Complexity[2]. MDL can be seen to be a con-

[2]Solomonoff mentioned Van Heerden, 1963, "A General Theory of Prediction",

strained version of the Kolmogorov Complexity in the sense that the reformulation no longer needs to be in terms of Universal Machines. Instead the reformulation is over a smaller class such as the class of polynomials etc..

The MDL principle states that data should be described in the shortest possible fashion (using a class of functions not necessarily a fully fledged Universal Machine). To understand the following, we will first describe a historical interpretation of this. An approximation of the MDL principle used often in the literature is,

$$\arg\min_{\delta,\theta} \left[L(x) = L(x|\theta) + L_\delta(\theta) \right], \tag{5.18}$$

where x are the data points, and θ are the parameters identifying a model given a model class up to a given precision δ, $L_\delta(\theta)$ is the number of bits used to code the parameters, and $L(x|\theta)$ is the number of bits used to code the residual. The well know version is the Maximum A Posteriori equivalent, where the code-lengths are calculated through Shannon's relation $L(y) = -\log P(y)$ (Shannon and Weaver, 1949; Wiener, 1948).

As indicated in (Rissanen, 1989, p. 58) this particular version is usually *not* the shortest possible. Take an example of the class of polynomials and an assumed normal distributed error function. For each polynomial all possible data sets have a non-zero probability and hence a code length, i.e. a particular data set has infinitely many code-lengths. To reduce the actual code-length one could therefore restrict the codes to complete codes where each data set only has a single code-length.

Several attempts to derive concrete algorithm achieving this improvement (for small sets of data) have been proposed (Clarke and Barron, 1990; Nohre, 1994; Rissanen, 1996; Dom, 1996), where Rissanen and Dom's approaches are similar and will be discussed in the following. The improved scheme suggests that given a model estimator, only a limited number of data sets will result in the exact same

Polaroid Corp. Cambridge 39, Mas. (Privately circulated report) as the inventor of this principle (Solomonoff, 1964, p. 254), and Wallace and Boulton (Wallace and Boulton, 1968) have proposed the Minimum Message Length (MML) principle which is similar (Baxter and Oliver, 1994)

model, and we are thus able to refine the functional by normalising the error code with respect to this restriction,

$$L(x) = L^*(x|\theta) + L_\delta(\theta) = -\log \frac{P(x|\theta)}{\int_\Omega P(y|\theta)\, dy} - \log P(\delta\theta), \quad (5.19)$$

where Ω is the set of data sets, y, yielding the same estimation point θ. Rissanen has shown that when using Jeffrey's prior this is equal to (Rissanen, 1996, eqn. 6),

$$L(x) = -\log P(x|\hat{\theta}) + \frac{k}{2}\log\frac{n}{2\pi} + \log\int\sqrt{|I(\theta)|}\, d\theta + o(1), \quad (5.20)$$

where $\hat{\theta}$ is the maximum-likelihood of the parameter on the data, k is the number of parameters, n is the size of the data set, and $I_{ij} = \frac{\partial^2 \log P(x|\hat{\theta})}{\partial\theta_i\partial\theta_j}$ is the Fisher information matrix.

In (Dom, 1996) this code-length functional has been evaluated for the Gaussian error and the general linear regression model class. It is generally found (Dom, 1996, corrected version of eqn. 66) that,

$$L(x) = n\log\hat{\sigma} + \log\frac{4(\pi n)^{n/2}\left[\left(\frac{d}{2\hat{\sigma}_0}\right)^k - 1\right]}{k^2\Gamma(\frac{k}{2})\Gamma(\frac{n-k}{2})}, \quad (5.21)$$

where Γ is the gamma function, $\hat{\sigma}$ is the maximum likelihood estimate of the standard deviation, $\hat{\sigma}_0$ is the quantisation constant given by the data set, and d is the range of values to be considered. Rissanen also offers an approximation of the same functional for the Gaussian error (Rissanen, 1996, eqn. 40) as,

$$L(x) = \frac{n}{2}\log\frac{2\pi e\sigma^2}{\delta} + \frac{k}{2}\log 2|\Sigma| + s(k-1) + \log\frac{C(k-1)}{k-1} + \log^* r + \log^* s + o(1), \quad (5.22)$$

which increase by the factor $r(k-1)$ for variances less than 1. δ is a constant related to the conversion of densities into probability

functions, Σ is the covariance matrix of the regression variables, s is the least integer such that $2^{2s} \geq \theta^T\theta$, r is the least integer such that $2^{-r} < \sigma$ (the ML estimated standard deviation of the error), and finally $C(l)$ is the volume of a l dimensional ball. This last functional is the one we will use because of its relative computational simplicity over Dom's functional. One should note, that Equation 5.22 is written in terms of density functions and *not* probability functions. To achieve the correct code-lengths, δ must be chosen in an intelligent fashion such that at least $\delta/\sigma^2 > 2\pi e$. For this purpose we will calculate $\frac{n}{2}\log\frac{2\pi e\sigma^2}{\delta}$ as $nH\{G_{\delta y}\}$, where H is the entropy of a discrete approximation of the normal distribution discretized to the same precision as the data. A similar argument could be made for the covariance matrix of the regression variables, but we will assume that the determinant is larger than 1.

Under the restriction that the contour between knots must be 1D functions, we are now able to decide on the number and placement of knots and the number of parameters, we need to model the contour in between knots, in order to describe a blob.

Either one of Equation 5.20-5.22 is highly non-linear and there is no guarantee that an optimal solution can be found. A greedy algorithm has been implemented which utilise an initial code for the model and the residuals. This algorithm is initialised with a large set of knots and at each iteration the knot yielding highest code length reduction is removed, until no removal yields a coding reduction. This in turn yields a set of frequencies of the models and residuals which in the following will be analysed to tailor specific codes.

We have described a blob as a list of knots and a polynomial representation connecting adjoining knots into a closed, non-intersecting curve. We will split the coding problem in two. First the knots will be coded/send followed by the polynomial descriptions. I.e. the description length is calculated as the sum,

$$L = L_{\text{knots}} + L_{\text{monge}}. \quad (5.23)$$

The knots are coded by Elias' codes (Rissanen, 1989) and the Monge parameters by the implied codes of Equation 5.22.

We are now ready to complete our algorithm from section 5.3.4:

5. For each polynomial piece between knots, find the optimal MDL degree from the class of models with or without the knot based models.

6. Iteratively remove the knot yielding largest reduction in the total coding cost until no further decrease can be found.

5.5 Coding an Alphabet

We have modeled 158 blobs by their contours as a list of knots and a 2 parameter polynomial representation per knot with a total of 1193 polynomial pieces and an equal number of knots, which is an average of about 7.5 pieces per blob with the use of a total of 55957 bits (as estimated by the MDL functional). A representative selection is shown in Figure 5.12. The top row shows the representations for small blobs, in the second row models with very good correspondence are shown. In the third row, models for italic characters are demonstrated, and in the last row, a selection of models with the worst experience fits are shown.

The model for the blob 'e' (lowest left in Figure 5.12) demonstrates the effect of setting the lowest scale level. Two severe noise instances are present. At the leftmost edge two pairs of pixels are flipped. This is ignored by the contour algorithm, since it is on a detail level below the lowest scale. Another noise instance occurs in the hole of the 'e'. This detail is above the lowest scale level, hence the hole is coded as two contours. Note that for the left part of the hole, the model is seen to be a line. This is not an error but implies that this hole is coded solely as noise.

The setting of the lowest scale level does also effect the amount of cornerness that can be modelled as demonstrated for the blob 'é' (lowest right in Figure 5.12). Here the accent has melted together with the 'e' creating a very sharp corner. Linear Scale-Space erodes sharp corners very quickly, resulting in a contour which is more blunt. This effect is always present when using two dimensional operators to estimate derivatives. It is not considered a major problem for this alphabet.

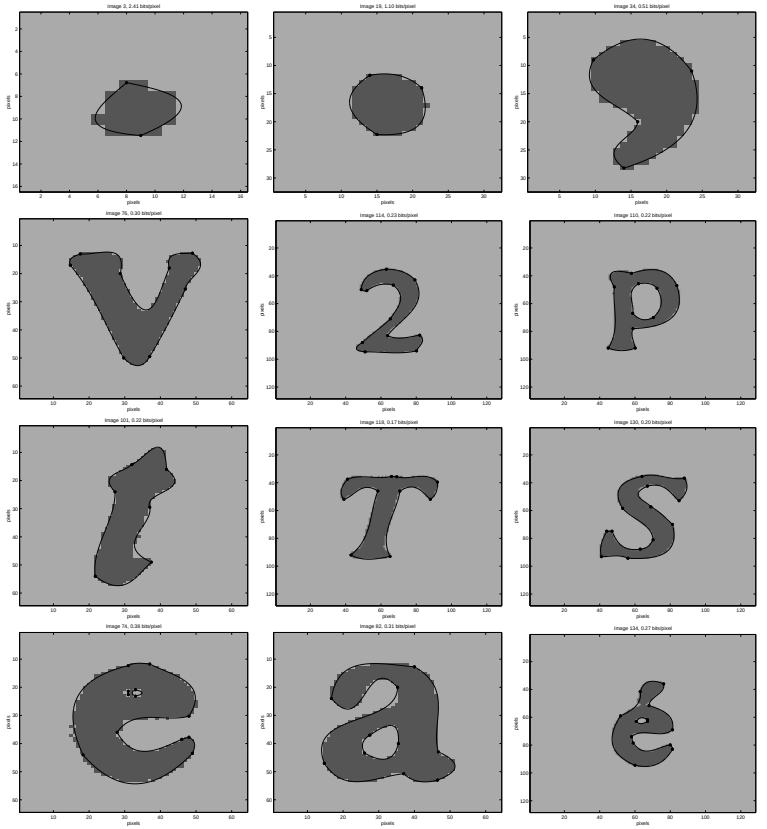


Figure 5.12: The top row shows the representations for small blobs, in the second row models with very good correspondence are shown. In the third row, models for italic characters are demonstrated, and in the last row, a selection of models with the worst experience fits are shown.

For a collection of 157 blobs from a fax-page, we observe that the model uses 3-4 knots to code a circular boundary, where 4 knots are used for large circles. There is a tendency that straight pieces are not coded as such but we conclude that this is only natural since straight pieces are not a generic part of the 2 parameter polynomial model class. Be reminded that 2 parameters imply a polynomial of 3rd degree. Finally, the models resulting from the described optimisation scheme is judged as being good.

We will at this point restrict ourselves to investigate the frequency data of the various aspects of the contour description which will be presented in the following.

5.5.1 A Code for Knots

It is immediately clear that the knots should not be coded by their absolute value as e.g. indirectly suggested by Banerjee *et al.* (Banerjee et al., 1996). Alternatively one could represent them as displacement vectors from the center of gravity, or what we will prefer, relative displacement with respect to each other in a sequential manner. The last two representations allow for a clear utilisation of the fact that the contours are closed and hence there will be a tendency for the angular change between two knots to be skewed due to the fact that the integral of the curvature of a closed curve is always 2π. We will compare the following two representations of the knots.

- Cartesian codes for the displacements.
- Polar representation of the displacements.

Cartesian codes are easily implementable in terms of choosing the precisions, which are identical for all coordinates. Codes for Cartesian coordinates we will investigate are Elias' code (Elias, 1975) (or Rissanen's Universal Prior of Integers (Rissanen, 1989)), which is suspected to be optimal due to Benford's law (Buck et al., 1993) and exponential distributions that have the advantage of being parameterisable. The polar representation, uses the length and angle parameters between consecutive knots which will allow for a utilisation of the expected

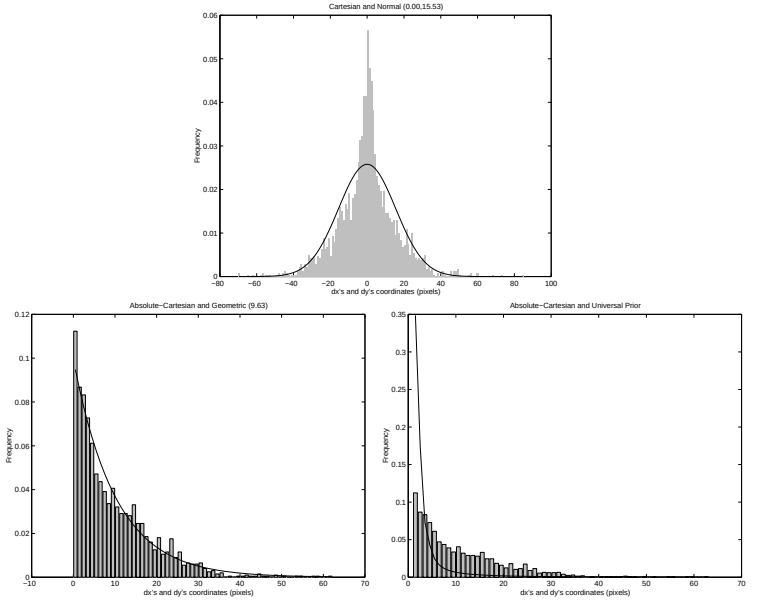


Figure 5.13: Frequency data for the vectors connecting the knots in each closed contour. TOP: a Gaussian model, BOTTOM LEFT the absolute coordinates with a geometric model, and BOTTOM RIGHT a hyperboloid (Rissanen's Universal Prior) model.

skewness in the distribution of the angles. The length parameter will probably be close to log-normally distributed, while the angle is less likely to correspond to simple distributions. Further, the truncation of the length is like in the Cartesian case simple, but angular precision required will be proportional to the length, hence a bit more complicated.

For the implemented MDL functional the distribution for the Cartesian representation together with three models are shown in Figure 5.13. The frequencies for the polar representation together with a model for the lengths are shown in Figure 5.14.

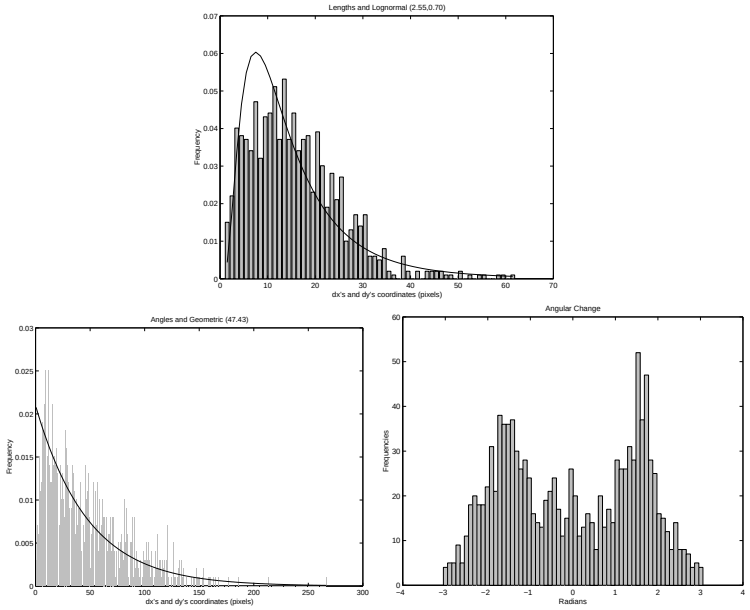


Figure 5.14: Frequency data for the vectors connecting the knots in each closed contour in polar form. The lengths (TOP) are nicely log-normal distributed, and the angular changes (BOTTOM LEFT) are distributed geometrically after truncation. Before truncation the angular changes are clearly bimodal (BOTTOM RIGHT). This property has been investigated, but the coding improvement is not noteworthy.

Form	Code and Parameters	Code length
Cartesian	Entropy	13952.75 bits
	Normal (0.0,15.5)	14344.92 bits
	Exponential (10.9)	14083.54 bits
	Geometric (11.04)	14086.74 bits
	Universal Prior	18100.63 bits
Polar	Log-normal (2.7,0.7)	6544.29
	and Geometric (54.6)	+8627.09 = 15171.38 bits

Table 5.1: A comparison of the bit cost using different representations and coders for the knots. The exponential, geometric, and the universal prior are calculated on the absolute values (absolute plus one for the universal) adding one bit for the sign per displacement. The Polar angle is calculated to $1/l$ precision, where l is the length of the particular displacement. Some of the above distributions are parameterised. These are first sent by Universal Prior code to one decimal point precision.

Of the before mentioned 55957 bits the knots coded are assumed to be coded using Elias' code or equivalently to be distributed according to Rissanen's Universal Prior of Integers. As demonstrated this is not the optimal code. In Tables 5.1 and 5.2 are given a comparison of the above described model distributions in terms of the resulting code lengths at a precision of 1pixel^2. The longest displacement vector may be inferred since all contours are closed. Also, the absolute displacement of the blobs can be inferred since the bounding box does not include a white boarder.

5.5.2 A Code for Polynomial Parameters

The coding of the parameters for the polynomial pieces is a little more tricky. For one thing, it is unavoidable not to have a truncation depending on the length of the polynomial arc. We are only going to investigate the coding of the polynomial parameters directly although we note that this information may also be coded as the slope of each

Form	Code and Parameters		Code Length
Cartesian	Entropy		11278.77 bits
	Normal (0.0,13.1)		11512.67 bits
	Exponential (8.9)		11378.76 bits
	Geometric (9.6)		11372.19 bits
	Universal Prior		14725.18 bits
Polar	Log-normal (22.6,0.7)	5204.45	
	and Geometric (47.4)	+7006.22 =	12210.67 bits

Table 5.2: The same data as Table 5.1. Here the longest displacement has been ignored since it can be reconstructed by the knowledge that the contours are closed.

knot.

Given two knots the polynomials are defined as,

$$f(x) = (x - x_n)(x + x_n)(ax + b), \tag{5.24}$$

where the coordinate system is aligned with the line joining the two consecutive knots with the zero point exactly midway. Since we have restricted ourselves to only use polynomials that go through the knots, we only have to consider the coding of the (a, b) pair.

The truncation issue will be determined by the following condition,

$$\max_x l(x) = \max_x \left| \begin{pmatrix} \partial f(x)/\partial a \\ \partial f(x)/\partial b \end{pmatrix} \right| \leq \delta, \tag{5.25}$$

where δ will be taken to be 1. I.e. we will truncate such that the maximum difference at any point along the curve will be less than a pixel. This implies,

$$l(x) = |x^2 - x_n^2| \left| \begin{pmatrix} x \\ 1 \end{pmatrix} \right| = |x^2 - x_n^2| \sqrt{x^2 + 1}. \tag{5.26}$$

The points of extremal change of this length with respect to x are the points $x \in \{0, \pm\sqrt{x_n^2 - 2}/\sqrt{3}\}$ of which 0 is a maximum when

$0 < x_n < \sqrt{2}$, and $\pm\sqrt{x_n^2 - 2}/\sqrt{3}$ are maxima when $x_n^4 > 2 + x_n^2$. I.e. below $\sqrt{2}$ we have a single maximum, and above there are two maxima. It is safe to assume that for polynomial pieces where the distance between two knots is less than $2\sqrt{2}$, the MDL optimisation will with very high probability choose a polynomial of degree 0. Hence we will concentrate on $x_n > \sqrt{2}$. This implies that the sensitivity to truncation as a function of x_n is given as,

$$l(\pm\frac{\sqrt{x_n^2 - 2}}{\sqrt{3}}) = \frac{2\sqrt{(1 + x_n^2)^3}}{3\sqrt{3}}. \tag{5.27}$$

Hence, the truncation should be chosen such that,

$$|(\delta a, \delta b)^T| \leq \frac{\delta 3\sqrt{3}}{2\sqrt{(1 + x_n^2)^3}}. \tag{5.28}$$

Following Nohre (Nohre, 1994) we view a and b as parameters of an orthogonal system and we will thus assume equal truncation: $\delta a = \delta b = \frac{\delta 3\sqrt{3}}{2\sqrt{2(1+x_n^2)^3}}$.

The distributions for the parameters extend very far and are very leptokurtic. This makes a difficult distribution to code. And for the same reason, we will not present any graphs of the distributions.

In Table 5.3 is the cost of coding the parameters jointly, in Tables 5.4 and 5.5 are given the cost of coding each first and second parameter separately. We conclude that the Universal Prior is best suited to code the truncated parameters and that there is no need to partition the description into two distributions.

5.6 Blob Coding in Perspective

Models for describing blobs are a central issue in applications related to image storage or transmission. In this work, a novel model class has been suggested taking the one dimensional nature of blob borders into account, yielding both compact codes and good descriptors. Optimisation is very much a part of finding good models within a class, and

Form	Code and Parameters	Code length
Cartesian	Entropy	11784.76 bits
	Normal (5.5,170.2)	22599.78 bits
	Exponential (167)	20717.70 bits
	Geometric (33.47)	17945.95 bits
	Universal Prior	14663.43 bits

Table 5.3: Estimated coding cost of discrete but ideal codes for the total number of parameters.

Form	Code and Parameters	Code length
Cartesian	Entropy	3955.37 bits
	Normal (0.0,47.5)	9105.58 bits
	Exponential (46.9)	8117.11 bits
	Geometric (7.47)	6439.46 bits
	Universal Prior	5216.63 bits

Table 5.4: The same estimation procedure as in Table 5.3 but for the first parameter.

Form	Code and Parameters	Code length
Cartesian	Entropy	7449.76 bits
	Normal (11.0,235.9)	11879.56 bits
	Exponential (228.6)	11012.16 bits
	Geometric (59.5)	9967.15 bits
	Universal Prior	9448.41 bits

Table 5.5: The same estimation procedure as in Table 5.3 but for the second parameter.

here we have presented a greedy thinning algorithm applied on a carefully selected set of knots. We judge from the experiments that this algorithm is a good tradeoff between compression time and rate, and we conclude that for large blobs this model class is a good competitor to the algorithm CONTEXT.

Coding the knots first has the distinct advantage that in the choice of knot placements, there lies a definite statement about the shape. I.e. the relative relationship between knots gives a crude description of the shape and thereby its curvature. Two models solely based on a sequence of knots immediately come to mind:

- A two dimensional cubic spline through each knot.
- A local estimation of the slope at each knot.

The local approach is preferable for several reasons. Firstly we believe that the structure is locally determined, i.e. large extrema in the curvature which are likely candidates for optimal knot placements structurally distinguish regions on the curve. Secondly, a strictly local approach will be faster to compute. Finally we acknowledge that the blobs we will describe will have discontinuities in the curvature function.

In Figure 5.15 is shown how we can calculate a local estimation of the slope as the angular mean between the two lines connecting three consecutive knots. Since Linear Scale-Space makes everything smooth, we may infer that the closer the knots are, the better this model will be. Conversely, if the knots are far apart, it is less likely that this model is good, hence we will allow for a coding, indicating if the model is to be used or not at the cost of one bit per segment. Assume that we will optimise over polynomials of degree 0-3. This yields 8 different polynomial descriptions: polynomials of 0-3 degree with or without the above model. The optimisation problem is a little simpler though. We require the polynomial to pass through the knots fixing two degrees of freedom leaving 6 possible choices: the zero function or 2nd or 3rd degree polynomial all with or without the knot based models. It should be noted that this spline model is similar to the Catmull-Rom splines, but not identical.

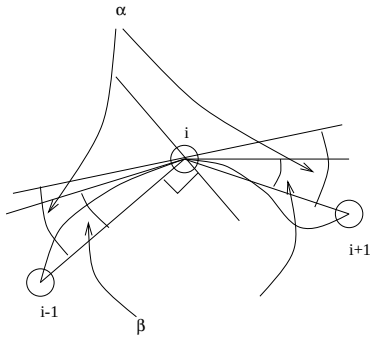


Figure 5.15: The modeled and the actual derivatives at a knot. 3 consecutive knots are connected with straight lines. The derivatives for the Monge patches (the smoothly varying curves) at knot k are modeled as α being half the angular change between the two straight lines. The Monge constraint restricts the value $\alpha - \beta_j$ to the closed interval $(-\pi/2, \pi/2)$.

From a few experiments it is estimated that with the use of the above spline models and increasing the optimisation space form 2 parameters to $\{0, 1, 2\}$ parameters it will be possible to decrease the total coding length with approximately 20%.

To end, this work has shown that it is feasible to have analytical representation of blobs, and that is is possible to estimate such a representation from bitmaps. Analytical representations are useful in several ways. If, for instance, the blob is to be decoded at another resolution than the original. Although it is not without problems to go to finer resolution for a number of the shapes in the examined alphabet, it is certainly aesthetically possible. I.e. one might interface such a model between bitmaps at low resolution and printers with high resolution. It might also be feasible to derive a resolution dependent description on the described model class. This could be useful if the blobs are to be decoded in a coarse to fine manner e.g. in an Internet application. Finally, this model class may also be used to refine the

font technology used in the Postscript language in order to compress the very large font dictionaries.

5.7 Acknowledgments

We would very much like to acknowledge the following people for the pleasant discussions and great help I have gotten on this work: Ronald Arps (IBM), Corneliu Constantinescu (IBM), Jorma Rissanen (IBM), Byron Dom (IBM), Ole Fogh Olsen (University of Copenhagen), Mads Nielsen (University of Copenhagen), and Joachim Weickert (University of Utrecht/University of Copenhagen). This work would not have been the same without the deep insight of these people.

Part II

Theoretical Aspects

Chapter 6

Theoretical Aspects: Introduction

In Part I we saw the usefulness of scale-space and information theory in image processing. In the following chapters we will take a close look at information theory.

The basic function in information theory is the entropy. The entropy is a measure on the uncertainty on a stochastic source, where a stochastic source is a data generator described by the distribution of the output. I.e. the entropy is a function of the distribution of a stochastic source. A source for which one symbol has probability one and the rest zero, has the lowest uncertainty; it is statistically fully determined, what the symbols from the source will be. For this situation we set the entropy to be minimal. The opposite situation where every possible symbol is equally probable is most uncertain in its output, we set the entropy to be maximal. Such considerations led Shannon to define the entropy as (Shannon and Weaver, 1949):

$$S(\boldsymbol{p}(x)) = -\sum_{i=1}^{N} p_i \log p_i,$$

where $\boldsymbol{p} = (p_1, \ldots, p_N)^T$ is a discrete probability distribution[1].

We have already seen the use of entropy for modelling in Chapter 5. In this part we will investigate the effect of scale-space on the entropy function and related uncertainty measures. We will see that the generalization of the entropy is equivalent to a number of representations, and that this generalization carries a wealth of information about the source. Finally we will use the entropy to interpret a model selection algorithm.

6.1 Information measures in scale-spaces

In Chapter 7 we study the effect of scale-space on uncertainty measures like the entropy, and this is used in image processing for global scale-selection and size estimation, and to indicate new basic results on the gray-value histogram under a scale-space of the image.

A distribution is a measure of probability, and can be viewed as an image: Assume a source generating real numbers. We will never know the true distribution of the source, since we would need infinitely many numbers to measure the probability density. We have to suffice with an estimation of the density. This can be done by collecting the outcome of the source in a number of bins, i.e. the domain of real numbers is discretized and each discrete location integrates the outcome over the corresponding area. This is a process identical to taking a picture, and the chosen discretization is in the same sense arbitrary. It is thus natural to study all discretizations between the chosen and the worst using scale-space.

Applying scale-space to probability functions generates a family of functions that converges towards a constant function. In Figure 6.1 we have given an example of a one dimensional function embedded in linear scale-space. The figure shows both the function and the corresponding histogram of function values. The entropy is a measure of a distribution, and both the function and the histogram can be viewed as distributions through proper normalisation. Hence, to attribute a

[1] A discrete function $\boldsymbol{p} = (p_1, \ldots, p_N)^T$ is a discrete probability distribution if $p_i \geq 0$ for all i and $\sum_{i=1}^{N} p_i = 1$.

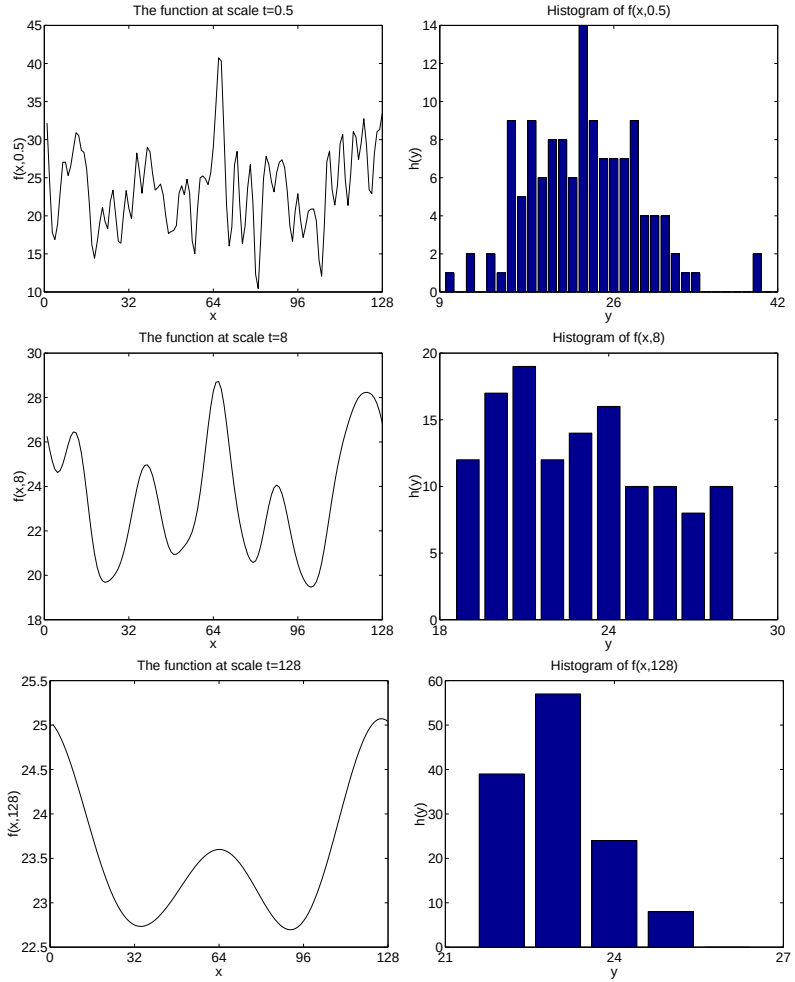


Figure 6.1: A function in scale-space and the corresponding histogram of function values. LEFT COLUMN: The function at various scales. RIGHT COLUMN: The corresponding histogram.

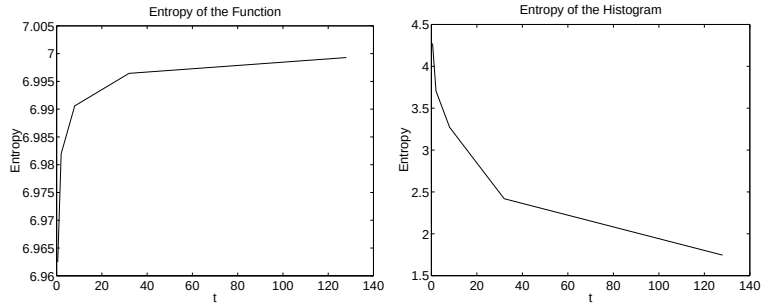


Figure 6.2: The evolutions of the entropy for two views of a function in scale-space. LEFT: The entropy when the function is viewed as a distribution. RIGHT: The entropy of the histograms.

single notion of complexity to the data we must choose a view of the function. In general the information content of a source is only defined up to its representation. In Figure 6.2 we show the evolution of the entropy of the two views. The entropy of the function seems to be increasing, while the entropy of the histogram seems to be decreasing. We know that when the scale is large, the function will be approximately constant. In terms of distributions, this implies that the function will converge to the uniform distribution, while the histogram converges to the Dirac delta[2] distribution. Hence, the entropy of the function will converge to the maximal entropy, while the entropy of the histogram will converge to the minimal entropy. In Chapter 7 we will show that the entropy of the function is monotonic. A similar result is not known for the evolution of the entropy of the histogram.

At a fixed scale, the entropy contains no information of spatial relations between function values. In that sense, the entropy of the function only depends on the histogram justifying the direct comparison done above. Both in information theory and image processing it

[2]The Dirac delta function $\delta(x)$ is defined by $\delta(x) = 0$, when $x \neq 0$ and $\int_{-\infty}^{\infty} \delta(x)\, dx = 1$

is of interest to extend the notion of entropy to that of generalized entropy. The direct result is that the spectrum of generalized entropies is *equivalent* with the histogram.

The generalized entropies are defined as (Rényi, 1976c; Rényi, 1976a; Rényi, 1976b):

$$S_\alpha = \frac{1}{1-\alpha} \sum_{i=1}^{N} p_i^\alpha .$$

The parameter α is called the information order. The generalized entropies are not defined for $\alpha = 1$, but by l'Hôpital's rule we see that the generalized entropies converge to the entropy at this point. Thus the entropy is considered part of the generalized entropies. For negative information orders, the generalized entropy is only defined when the distribution is larger than zero everywhere[3]. While this is a problem for distributions in general, all distributions embedded in scale-space fulfil this restriction except at scale zero.

In Figure 6.3 is given an example of a random function, its histogram, and the generalized entropy for positive orders. It can be proven that the generalized entropy is a decreasing function of order, as confirmed by the figure. For a fixed scale the generalized entropies are independent on the spatial relation between probability values. However, the evolution of the generalized entropies, when the function is embedded in scale-space, is strongly restricted. One basic result obtained is that also the generalized entropies are monotonic in scale.

In the linear scale-space we may use the generalized entropies to perform size analysis of image structure. This can be done since the change of the entropies will depend on the size of the Gaussian kernel and the size of image structure. In Figure 6.4 is given an example. A blob is shown together with two circles denoting kernels of two different sizes. Consider the image of the blob smoothed by the small kernel. A slight change of the kernel size will not alter the result significantly. Likewise, for an image of the blob smoothed by the very large kernel. A slight change of the kernel size will not alter the result

[3] Alternatively, it is possible to define $\tilde{S}_\alpha = \frac{1}{1-\alpha} \sum_{i|p_i>0} p_i^\alpha$, which is defined for all α.

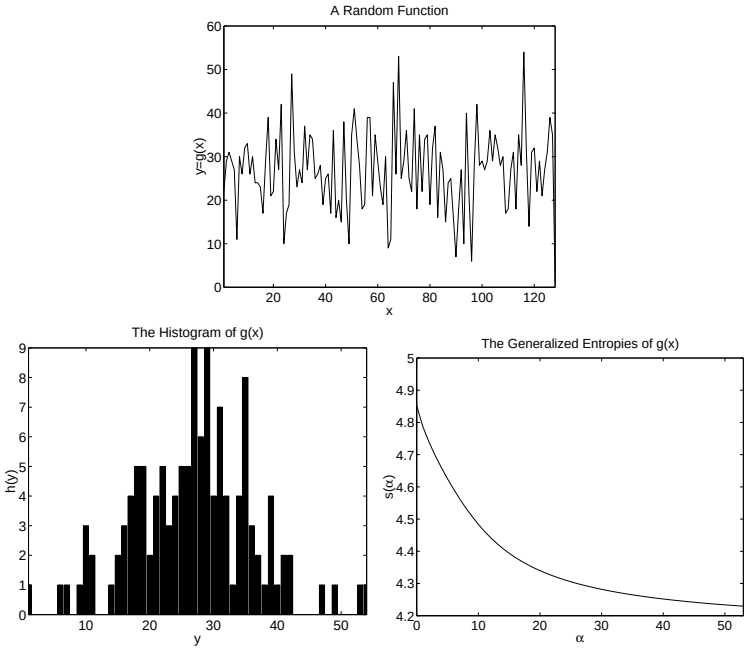


Figure 6.3: A random function normally distributed (TOP), its histogram (BOTTOM LEFT), and its generalized entropy for positive order (BOTTOM RIGHT).

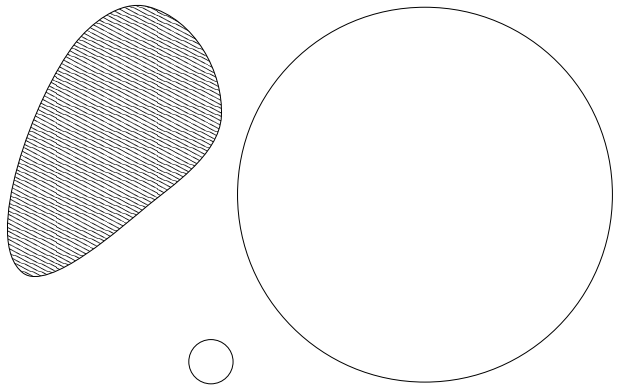


Figure 6.4: A blob (DASHED) and two Gaussian kernels of different size denoted by circles.

significantly. The smoothing results using the small and the large kernel are, however, significantly different images, and we can conclude that the change must have taken place somewhere between small and large kernel size. In fact, we expect that the change of kernel size will have the largest effect, when the kernel size is approximately that of the image structure. This effect must be visible in the change of the generalized entropies. Chapter 7 demonstrates the effect and usage in image processing.

The work on generalized entropies gave inspiration to take a closer look at histograms, which will be discussed in the folowing.

6.2 Some theorems on continuous histograms

The discrete histogram of images is not only intimately linked to the generalized entropies as described above, but also to the multifractal spectrum and the spectrum of moments. The precise relation is given

in Chapter 7.7. The work presented in Chapter 8 is strongly motivated by these relations.

In Chapter 8 we study the continuous histogram of a one dimensional function, which may seem like a severe limitation of the above relations, but we consider the continuous histograms as the first step in a deeper understanding of the discrete histograms. At least, a discrete histogram for a very finely sampled function shares some key aspects of the corresponding continuous histogram. In Figure 6.5 is shown a third degree polynomial at various samplings, and the corresponding histogram. We note that the discrete histogram seems to have a pole structure corresponding to the extrema of the function. Such considerations lead us to use the folowing definition of a continuous histogram for a C^1 function $g(x)$.

$$h(y) = \sum_{x:g(x)=y} \frac{1}{|g'(x)|}.$$

Here $g'(x)$ denotes the derivative of g with respect to x. Using this definition we may calculate the continuous histogram for the polynomial in Figure 6.5 as shown in Figure 6.6. As discussed in the previous section, the discrete histogram is oblivious of the spatial relations between function values. The main result of Chapter 8 is that this is definitely not the case for continuous histograms. We prove that for a large class of functions the continuous histogram uniquely specifies a function up to translation and mirroring of the domain. While we have not been able to prove this for all functions, we can show that severe constraints on all functions exist by their continuous histograms.

6.3 On the invariance of saliency based pruning algorithms

Finally, in Chapter 9, we revisit the subject of model selection by information theory. In contrast to Chapter 5 we will study a model selection algorithm that can be interpreted in terms of information theory.

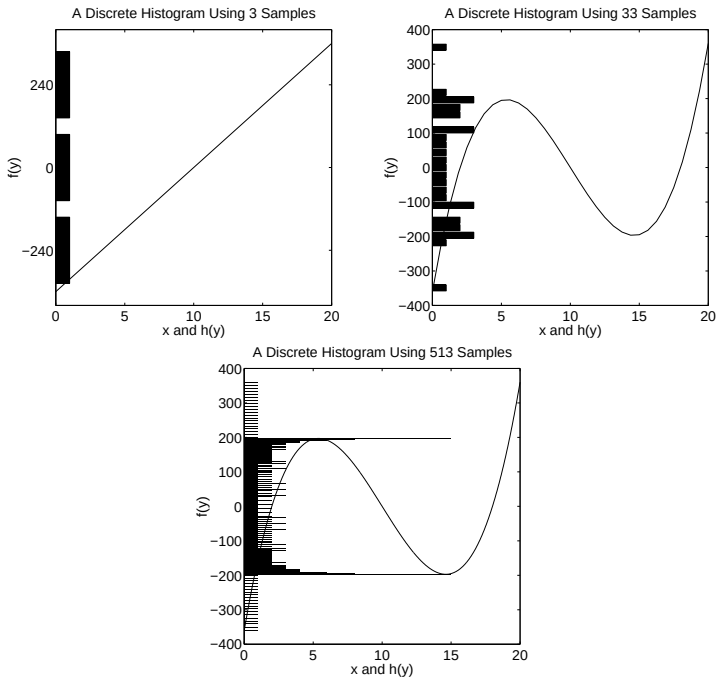


Figure 6.5: A function and its histogram under different sampling rates. The function $f(x) = (x-2)(x-10)(x-18)$ is shown using 3, 33, and 513 sampling points on the horizontal axis. The histogram is shown using the same sampling rates as a projection onto the vertical axis. The interference between these two samplings causes the Moiré patterns noticeable in the bottom graph.

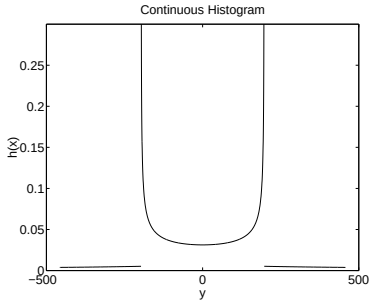


Figure 6.6: The continuous histogram of the polynomial $f(x) = x^3 - 64x$.

In 1990 (Cun et al., 1990), it was suggested to reduce the complexity of a function by examining an error measure by its lower order terms in a Taylor series. The field of application was that of feed-forward neural networks, but this shall not concern us further. The basic idea was that for models with a high number of parameters compared to the size of the dataset being analysed, the number of parameters could be reduced (explicitly be set to zero) according to an analysis of saliency. That is, using some measure on the difference between the function and the dataset, the effect of explicitly setting a parameter to zero could to sufficient accuracy be estimated by the lower order terms of a Taylor series of the measure. The algorithm Optimal Brain Damage (Cun et al., 1990) hence suggest an ordering of the parameters by their saliency, and to reduce the complexity by removing the least salient parameter.

This seems like a very general technique, but also seems to fail to take into account the complexity of the model class as emphasized by Minimum Description Length and Maximum A Posteriori techniques. Based on previous work we realized however that this failure is only apparent, and Chapter 9 demonstrates that any Taylor series used as described above will be invariant to certain functions of the model parameters. The key issue being that of symmetry in the Taylor extrapolation and the derivatives. As an example consider a model class

of just one parameter θ, and an analytical error measure E, hence also dependent on θ. The Taylor series of E is given by:

$$E(\theta+\delta) = \sum_{j=0}^{\infty} \frac{\delta^j}{j!} \frac{\partial^j E(\theta)}{\partial \theta^j}.$$

Setting θ to zero is equivalent to examining the above equation for $\delta = -\theta$. Thus if we add a function $F(\theta)$ to E which fulfils the following constraint:

$$\frac{\partial^j F(\theta)}{\partial \theta^j} = \frac{1}{\theta^j},$$

then the effect of the ordering obtained by a truncated Taylor series will be independent on F. Such functions will be independent on the dataset, and can in some cases be interpreted as a distribution on the parameter. In these situations, we may conclude that F represents the complexity of the model class, or equivalently the implicit prior of Optimal Brain Damage.

Chapter 7

Information Measures in Scale-Spaces[1]

7.1 Introduction

In recent years multiscale techniques have gained a lot of attention in the image processing community. Typical examples are pyramid and wavelet decompositions. They represent images at a small number of scales and have proven their use for image compression in numerous implementations. Another important class of multiscale techniques consists of so-called *scale-space representations* (Iijima, 1962; Weickert et al., 1997a; Witkin, 1983; Koenderink, 1984). They embed an original image into a continuous family of subsequently simpler versions. Many scale-spaces can be formulated as the evolution of the initial image under a suitable linear or nonlinear diffusion process. Such an image evolution is useful for tasks such as feature extraction, scale selection, and segmentation, see (Lindeberg, 1994; Haar Romeny, 1994; Sporring et al., 1997) and the references therein.

[1] An earlier version of this chapter has been published in a conference proceeding (Sporring and Weickert, 1997). The current version is resubmitted for a journal publication as: Jon Sporring and Joachim Weickert, "Information Measures in Scale-Space".

Besides multiscale ideas, also information theoretical concepts such as the Shannon–Wiener entropy (Shannon and Weaver, 1949; Wiener, 1948), Rényi's generalized entropies (Rényi, 1976c; Rényi, 1976a; Rényi, 1976b), and the Kullback–Leibler distance (Kullback and Leibler, 1951) have made contributions to image analysis; for instance Brink and Pendock (Brink and Pendock, 1996), Brink (Brink, 1996), and Sahoo *et al.* (Sahoo et al., 1997) have used them for local image thresholding, and Vehel *et al.* (Véhel, 1998) and Chaudhuri and Sarkar (Chaudhuri and Sarkar, 1995) study images in a multifractal setting. It is not difficult to see that the generalized entropies, the multifractal spectrum, the gray-value moments and the gray-value histogram itself are equivalent representations: they can be transformed into each other by one-to-one mappings. More details can be found in Section 7.7.

Since scale-spaces simplify images, it is only natural to investigate their simplification properties in terms of information measures. Already in 1949, Shannon mentioned that the Shannon–Wiener entropy decreases under averaging transformations(Shannon and Weaver, 1949, p. 52). In 1993 Illner and Neunzert (Illner and Neunzert, 1993) studied a biased diffusion process, where the original image evolves towards a background image b along a path where its Kullback–Leibler distance with respect to b increases monotonically. Jägersand (Jägersand, 1995) used the Kullback–Leibler distance in linear scale-space for focus-of-attention. Oomes and Snoeren (Oomes and Snoeren, 1996) used the entropy relative to a background measure to estimate the size of objects in images. Sporring (Sporring, 1996) applied the Shannon–Wiener entropy in linear scale-space to perform scale selection in textures and showed the monotone behaviour using concepts from thermodynamics. Weickert (Weickert, 1998) proved monotony of the Shannon–Wiener entropy in linear and nonlinear diffusion scale-spaces by regarding it as a Lyapunov functional. Lyaponov functionals have been used for scale-space synchronisation (Niessen et al., 1997) and for a uniform sampling of the scale axis with respect to its information content (Weickert et al., 1997b; Niessen et al., 1998). Peleg *et al.* (Peleg et al., 1984) used the fractal dimension in a morphological scale-space to study texture. Re-

lations between Shannon–Wiener entropy and multiscale concepts in terms of wavelets have been established by Krim and Brooks (Krim and Brooks, 1996), where inequality theory was applied to propose optimal measures for feature-directed segmentation.

The present paper extends previous work in this field by studying both theoretical aspects and the practical potential of generalized entropies in a linear and nonlinear multiscale setting. Generalized entropies are complete in the sense that they alow for a reconstruction of the gray-value histogram (see Section 7.7). A scale-space extension is used to complement the entropies with spatial information. We prove monotony and smoothness properties with respect to the information order and the scale parameter. We use the scale-space behaviour of generalized entropies for scale selection and size estimation, and we introduce a fingerprint-like description for textures. The results indicate that our extensions broadens the potential use of entropy methods in image analysis. Some preliminary results in this paper have been presented at conferences (Sporring, 1996; Sporring and Weickert, 1997).

Throughout this paper we identify an image by its two-dimensional distribution of light on a rectangular image domain. It should be noted that this representation is invariant under multiplication with, but not under addition of, a constant. It is important to note that this two-dimensional distribution is *not* the gray-value histogram.

The outline of this chapter is as folows. In Section 7.2 will be given a brief introduction to linear and nonlinear scale-spaces. Then in Section 7.3 we will investigate a scale-space extension of the generalized entropies. Finally in Section 7.4 we will describe some applications in image processing. A conclusion is given in Section 7.5.

7.2 A Short Introduction to Scale-Spaces

The images considered in this work are all discrete, but for simplicity we will in this section introduce two scale-spaces in the continuous setting. Discrete scale-space aspects are discussed by Lindeberg (Lindeberg, 1994) for the linear framework, and by Weickert (Weickert, 1998) for the nonlinear setting. Scale-spaces can be considered as an

alternative to traditional smoothing methods from statistics (Simonoff, 1996).

In scale-space theory one embeds an image $p(\boldsymbol{x}) : \mathbb{R}^2 \to \mathbb{R}$ into a continuous family $\{p(\boldsymbol{x}, t) \mid t \geq 0\}$ of gradually smoother versions of it. The original image corresponds to the scale $t = 0$, and increasing the scale should simplify the image without creating spurious structures. Since a scale-space creates a hierarchy of the image features, it constitutes an important step from a pixel-related image description to a semantical image description.

It has been shown that partial differential equations are the suitable framework for scale-spaces (Alvarez et al., 1993). The oldest and best studied scale-space obtains a simplified version $p(\boldsymbol{x}, t)$ of $p(\boldsymbol{x})$ as the solution of the linear diffusion process with $p(\boldsymbol{x})$ as initial value.

$$\partial_t p = \partial_{x_1 x_1} p + \partial_{x_2 x_2} p, \tag{7.1}$$

$$p(\boldsymbol{x}, 0) = p(\boldsymbol{x}), \tag{7.2}$$

where $\boldsymbol{x} = (x_1, x_2)^T$. It is well-known from the mathematical literature that the solution $p(\boldsymbol{x}, t)$ can be calculated by convolving $p(\boldsymbol{x})$ with a Gaussian of standard deviation $\sigma = \sqrt{2t}$:

$$p(\boldsymbol{x}, t) = (G_t * p)(\boldsymbol{x}), \tag{7.3}$$

$$G_t(\boldsymbol{x}) := \frac{1}{4\pi t} e^{\frac{-|\boldsymbol{x}|^2}{4t}}. \tag{7.4}$$

This process is called *Gaussian scale-space* or *linear scale-space*. It was first discovered by Iijima (Iijima, 1962; Weickert et al., 1997a) and became popular two decades later by the work of Witkin (Witkin, 1983) and Koenderink (Koenderink, 1984). A detailed treatment of the various aspects of Gaussian scale-space theory can be found in (Lindeberg, 1994; Florack, 1997; Sporring et al., 1997) and the references therein.

Unfortunately, Gaussian smoothing also blurs and dislocates semanticaly important features such as edges. This has triggered people to study nonlinear scale-spaces. Perona and Malik (Perona and Malik, 1990) proposed to replace the linear diffusion equation (7.1) by the nonlinear diffusion process

$$\partial_t p = \nabla \cdot (g(|\nabla p|)\, \nabla p), \tag{7.5}$$

where $\nabla = (\partial_x, \partial_y)^T$, and the diffusivity $g(|\nabla p|)$ is a decreasing function in $|\nabla p|$. The idea is to regard $|\nabla p|$ as an edge detector and to encourage interregional smoothing over intraregional smoothing.s,Thu locations where the gradient is large have a large likelihood of being an edge, and the diffusivity is reduced.

In our experiments we consider a nonlinear diffusion process where the diffusivity is given by (Charbonnier et al., 1994)

$$g(|\nabla p|) := \frac{1}{\sqrt{1 + |\nabla p|^2/\lambda^2}} \qquad (\lambda > 0). \tag{7.6}$$

Such a choice guarantees that the nonlinear diffusion filter is well-posed.

This is one of the simplest representative of nonlinear scale-spaces. Overviews of other nonlinear scale-spaces can be found in (Weickert, 1998; Haar Romeny, 1994).

7.3 Generalized Entropies

Let us now consider a discrete image $\boldsymbol{p} = (p_1, \ldots, p_N)^T$, where $p_i > 0$ for all i. Note that a single index is used for the two-dimensional enumeration of pixels. Its family of generalized entropies is defined a s,

$$S_\alpha(\boldsymbol{p}) := \frac{1}{1-\alpha} \log \sum_{i=1}^{N} p_i^\alpha \tag{7.7}$$

for $\alpha \neq 1$. The limit from left and right at $\alpha = 1$ is the Shannon–Wiener entropy,

$$S_1(\boldsymbol{p}) = -\sum_{i=1}^{N} p_i \log p_i, \tag{7.8}$$

and we might thus as well consider it as part of the continuum. The parameter α is called information order.

Let the vector-valued function $\boldsymbol{p}(t) = (p_1(t), \ldots, p_N(t))^T$ be the linear or nonlinear scale-space extension, where the continuous parameter

t denotes scale. These scale-spaces can be obtained by a spatial discretization of Equation 7.1 or 7.5 with reflecting boundary conditions.

We will now discuss some details of the mathematical structure of generalized entropies.

Proposition 7.1. *The generalized entropies are decreasing in α.*

Proof. Follows immediately from (Rényi, 1976c; Hentschel and Procaccia, 1983). □

Proposition 7.2. *The generalized entropies $S_\alpha(\boldsymbol{p}(t))$ are increasing in t for $\alpha > 0$, constant for $\alpha = 0$, and decreasing for $\alpha < 0$. For $t \to \infty$, they converge to the zeroth order entropy S_0.*

Proof. The proof is based on a result from (Weickert, 1998, Theorem 5): For a discrete image $\boldsymbol{p}(t)$, which is obtained from a spatially discrete diffusion scale-space, the following holds. The expression

$$\Phi(\boldsymbol{p}(t)) := \sum_{i=1}^{N} r(p_i(t)) \tag{7.9}$$

is decreasing in t for every smooth convex function r. Moreover, $\lim_{t\to\infty} p_i(t) = 1/N$ for all i.

Using this we first prove the monotony of S_α with respect to t. Let $\alpha > 1$ and $s > 0$. Since $r(s) = s^\alpha$ satisfies

$$r''(s) = \alpha(\alpha - 1)s^{\alpha-2} > 0, \tag{7.10}$$

it follows that r is convexs,Thu

$$\Phi(\boldsymbol{p}(t)) = \sum_{i=1}^{N} r(p_i(t)) = \sum_{i=1}^{N} p_i^\alpha(t) \tag{7.11}$$

is decreasing in t and

$$S_\alpha(\boldsymbol{p}(t)) = \frac{1}{1-\alpha}\log\Phi(\boldsymbol{p}(t)) \tag{7.12}$$

is increasing in t.
Similar reasonings can be applied to establish monotony for the cases $0<\alpha<1$ and $\alpha<0$.

For $\alpha = 1$ we obtain the Shannon–Wiener entropy for which monotony has already been shown in (Weickert, 1998).

Let $\alpha = 0$. Then

$$S_0(\boldsymbol{p}(t)) = \log\sum_{i=1}^{N} p_i^0(t) = \log N = \text{const.} \quad \forall\, t. \tag{7.13}$$

To verify the asymptotic behaviour of the generalized entropies we utilise $\lim_{t\to\infty} p_i(t) = 1/N$. For $\alpha \neq 1$ this gives

$$\lim_{t\to\infty} S_\alpha(\boldsymbol{p}(t)) = \frac{1}{1-\alpha}\log\sum_{i=1}^{N}\frac{1}{N^\alpha} = \log N = S_0, \tag{7.14}$$

and $\alpha = 1$ yields

$$\lim_{t\to\infty} S_1(\boldsymbol{p}(t)) = -\sum_{i=1}^{N}\frac{1}{N}\log\frac{1}{N} = \log N = S_0. \tag{7.15}$$

This completes the proof. □

The following smoothness results constitute the basis for studying derivatives of generalized entropies as will be done in Section 7.4.

Proposition 7.3. *The generalized entropies are* C^∞ *for* $\alpha \neq 1$ *and at least* C^1 *in* $\alpha = 1$. *For linear scale-space they are* C^∞ *in* t, *and for the nonlinear scale-space they are* C^1 *in* t.

Proof. In order to prove smoothness with respect to α, we first consider the case $\alpha \neq 1$. Then S_α is the product of the two C^∞ functions $\frac{1}{1-\alpha}$ and $\log\sum_{i=1}^{N} p_i^\alpha$, and thus also C^∞ in α.

The smoothness in $\alpha = 1$ is verified by applying l'Hôpital's rule. Straightforward calculations show that

$$\lim_{\alpha\to 1}\frac{\partial S_\alpha}{\partial\alpha} = \frac{\sum_{i=1}^{N} p_i(\log p_i)^2 - (\sum_{i=1}^{N} p_i\log p_i)^2}{2}. \tag{7.16}$$

Thus $\frac{\partial S_\alpha}{\partial\alpha}$ exists and S_α is in C^1.

For linear scale-space, C^∞ in t follows directly from the fact that $G_t(x)$ is in C^∞ with respect to t. In the nonlinear case, C^1 in t is a consequence of the fact that the solution $\boldsymbol{p}(t)$ is in C^1 with respect to t. This is proven in (Weickert, 1998, Theorem 4). □

Figure 7.1 illustrates the monotony of the generalized entropies both in scale and order for both scale-spaces. The figures have been created by finite difference algorithms which preserve the monotonic properties established in this section (Weickert et al., 1998).

7.4 Experiments

We will in this section demonstrate some applications for the generalized entropies in image processing. We will consider the change of entropies by logarithmic scale,

$$c_\alpha(\boldsymbol{p}(t)) := \frac{\partial S_\alpha(\boldsymbol{p}(t))}{\partial(\log t)}, \tag{7.17}$$

since this appears to be the natural parameter (at least for linear scale-space) (Koenderink, 1984),(Florack et al., 1992),(Lindeberg, 1994, section 8.7.1),(Sporring and Weickert, 1997). We emphasise that the generalized entropies are global measures and are thus best suited for images with homogeneous textures.

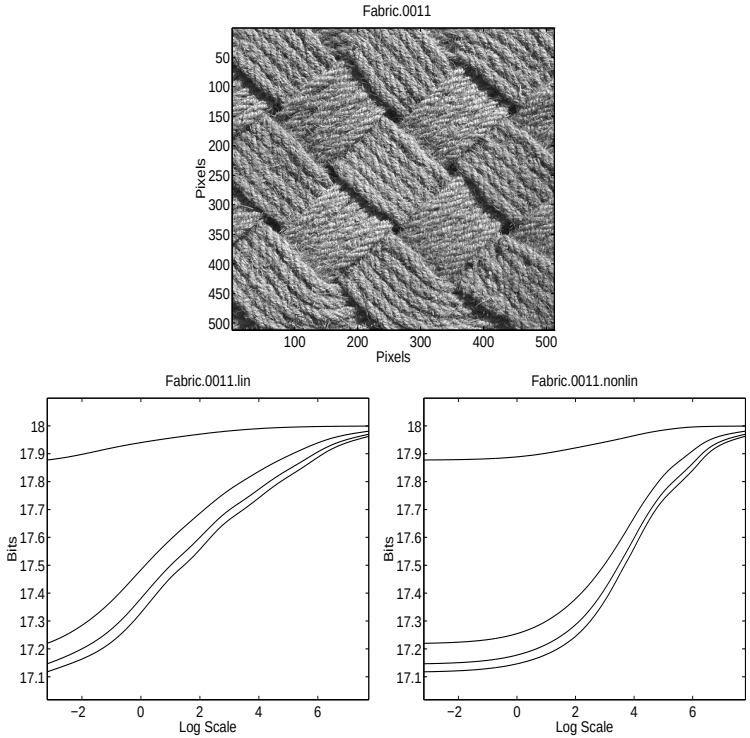


Figure 7.1: Examples of some generalized entropies. TOP: A 512×512 gray-valued image. BOTTOM LEFT: Generalized entropies in linear scale-space. From top to bottom $\alpha = 1, 34, 67, 100$. BOTTOM RIGHT: Ditto for nonlinear scale-space.

7.4.1 Shannon–Wiener Entropy and Zooming

This section analyses the zooming behaviour of the Shannon–Wiener entropy in linear scale-space.

Figure 7.2 (top left and right) shows images from a laboratory experiment: The camera is placed fronto-parallel to a plane with a simple texture: pieces of paper with discs arranged in a regular manner. A sequence is produced as a series of increasing zoom values. In Figure 7.2 (bottom) we plot the scale $\sigma = \sqrt{2t}$ of the point of maximum entropy change against the mean size of the discs. As can be seen the relation is close to linear. It appears that in linear scale-space the point of maximal entropy change by logarithmic scale corresponds to the size of the dominating image structures.

7.4.2 Spatial Extent of Structures

In this section we show that the scaling behaviour in linear scale-space carries over to the generalized entropies and that they can be used to simultaneously measure the size of light and dark structures. We shall also see that the latter cannot be done with the Shannon–Wiener entropy.

The idea is as follows: The definition of the generalized entropies implies that entropies for large positive α focus on high gray values (white areas), while for large negative value they analyse low gray values (dark areas).

We expect that $c_\alpha(\boldsymbol{p}(t))$ is especialy high for structures of diameter d, when the variance $\sigma^2 = t/2$ of the Gaussian is close to the variance of the structures. Let us for simplicity consider disc shaped structures. The second radial moment of a disc of diameter d is[2],

$$\sigma^2 = \int_0^{2\pi} \int_0^{d/2} \frac{r^2}{\pi(d/2)^2} r \; dr \; d\phi = \frac{d^2}{8}. \tag{7.18}$$

[2]After the defense I have realized that there is probably an error of a factor 2 in the equation and the experiment below: The second radial moment is equivalent to calculating the trace of the covariance matrix, and since the trace for an isotropic Gauss is $2\sigma^2$, equation 7.18 should probably be $2\sigma = \ldots$.

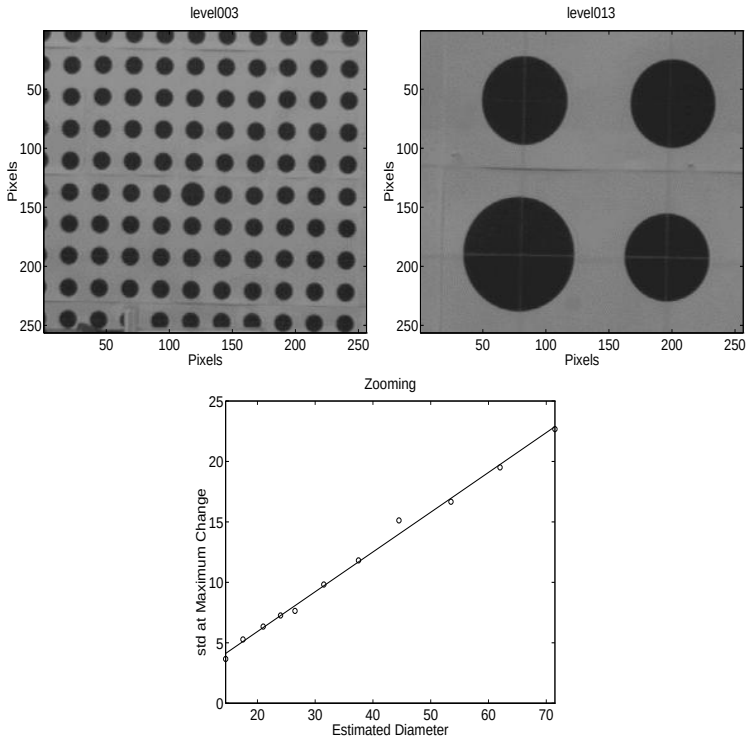


Figure 7.2: A zooming sequence. TOP: First and last image. BOTTOM: The $\sigma = \sqrt{2t}$ values maximising $c_1(\boldsymbol{p}(t))$ versus the estimated disc sizes.

Hence we expect a light (or dark) structure of diameter d to have a significant entropy change by logarithmic scale at time $\sigma^2/2 = d^2/16$. This size estimate remains qualitatively correct for non-disc structures. In this case, it gives the size of the largest minimal diameter.

Figure 7.3 shows the result of a performance analysis. The size estimate (7.18) has been applied to a number of simple sinusoidal images with structures (half wavelengths) between 1 and 257 pixels. As can be seen in the bottom graph, for sufficiently large structures the estimated sizes are close to the true size. Although by definition, the generalized entropies are not symmetric in order, both positive and negative orders have similar scaling behaviour which is close to linear.

In Figure 7.4 we show an experiment on a texture with a more complicated periodicity. This real image has been created by the Belousov–Zhabotinsky reaction (Jensen et al., 1998). From orders ± 20 we find dominating low intensity values corresponding to a diameter 7.9, while the dominating high intensity values suggest structures of diameter 3.7. From this we conclude that the distance between the light spiral arms in the mean is approximately 7.9 pixels, and the width of the spiral arms is approximately 3.7 pixels. In spite of the fact that the disc model (7.18) is not very appropriate for the line like structure, the size estimates are in the correct order of magnitude.

The Shannon–Wiener entropy cannot be used for size estimation since it is a mixture of information from both light and dark areas. Thus is does not allow for a distinction between fore- and background.

7.4.3 Fingerprints for Entropies in Scale-Space

Section 7.4.1 and 7.4.2 have shown that the scales of extremal entropy change carry significant information for selected information orders. Thus it would be interesting to introduce a compact description of the extremal changes for the continuum of information orders. In analogy with edge analysis in linear scale-space (Yuille and Poggio, 1986) we call such a description a fingerprint image. In Figure 7.5 are fingerprint images for two textures given, both in the linear and nonlinear scale-space. The fingerprint lines are the extrema of $c_\alpha(\boldsymbol{p}(t))$ in t. Our monotony results immediately imply the following consequences:

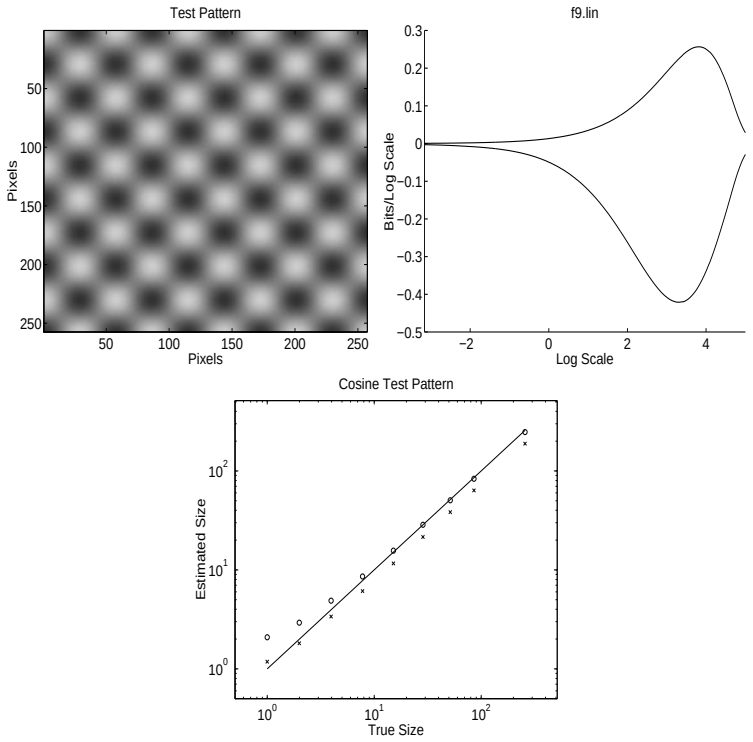


Figure 7.3: Scaling behaviour and size estimation with generalized entropies. TOP LEFT: Test image generated by $257^{-2}(1 + 0.6\cos(\omega x_1)\cos(\omega x_2))$ with $\omega = 9\pi/257$. TOP RIGHT: The corresponding $c_\alpha(\boldsymbol{p}(t))$ curves for $\alpha = \pm 100$. Top curve is for positive order and bottom curve for negative order. BOTTOM: A double logarithmic plot of the true size versus the estimated size for various ω. The straight line depicts the truth, the circles the estimation from order 100, and the crosses for order -100.

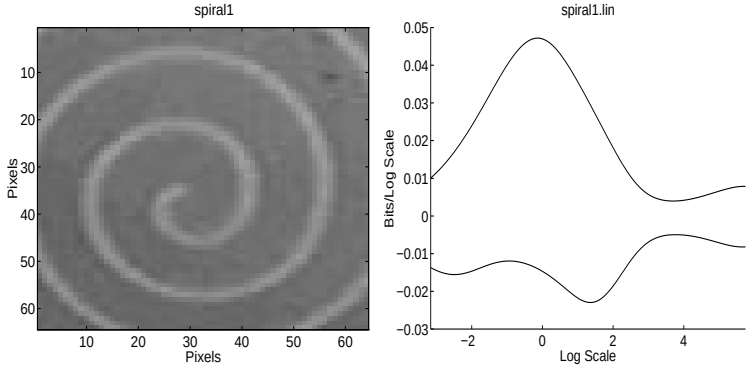


Figure 7.4: LEFT: Spiral generated by a chemical reaction. RIGHT: Entropy changes for orders 20 (top curve) and -20 (bottom curve).

If there is only one fingerprint line for a given positive order, then it corresponds to a maximum (likewise to a minimum for negative orders); see also Figure 7.3. For almost all orders there will be an odd number of fingerprint lines, which correspond to alternating maxima and minima. This can be seen for instance in the middle right graph in Figure 7.5. For information order 60, the leftmost line is a maximum followed by alternating minima and maxima.

It appears that the location of the fingerprint lines is more stable over information orders for the nonlinear scale-space than for the linear one. Due to the reduced diffusivity of the nonlinear scale-space, the fingerprint lines are shifted towards higher scales.

7.5 Conclusions

In this paper we have investigated entropies as a means for extracting information from scale-spaces. This has lead to the following contributions.

- Monotony and smoothness properties for the Shannon–Wiener

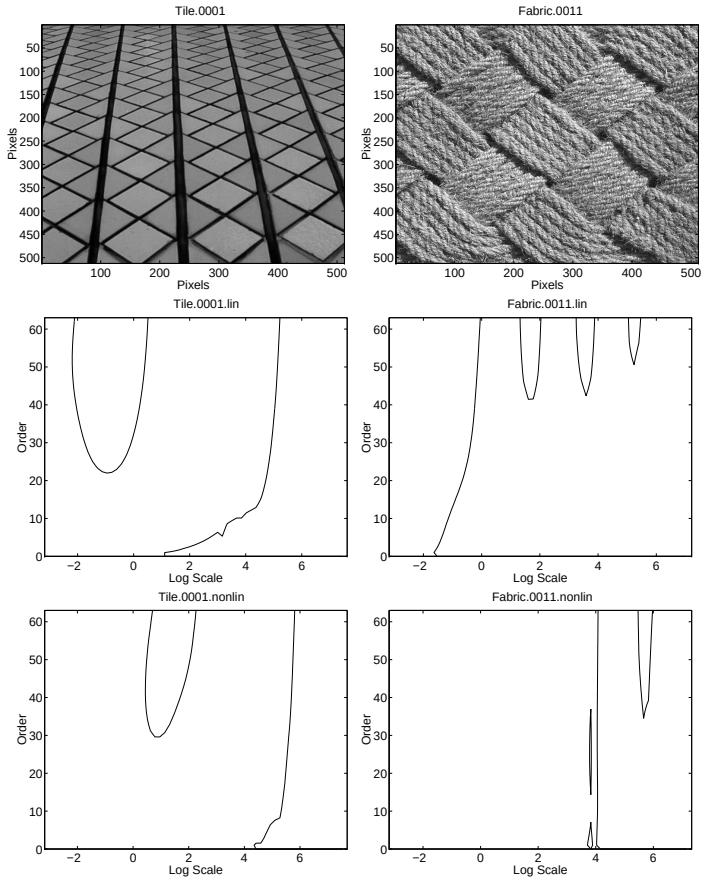


Figure 7.5: Fingerprints of generalized entropies. TOP ROW: Two textures. MIDDLE ROW: Fingerprints for linear scale-space. BOTTOM ROW: Ditto for nonlinear scale-space.

entropy and Rényi's generalized entropies have been proven for the linear and a nonlinear diffusion scale-space. The proofs hold also for all other nonlinear diffusion scale-spaces treated in (Weickert, 1998).

- We have illustrated that the generalized entropies can be used to perform size measurements for periodic textures. This is not possible with the Shannon–Wiener entropy. We have proceeded to define a fingerprint image for entropies in scale-space and analysed some of its basic properties. The localisation of the fingerprint lines can be improved using nonlinear instead of linear scale-space.

The following topics appear promising for future work.

- In the context of texture analysis, it would be interesting to perform an in-depth study on the relation between the fingerprint topology and the structure of the texture.
- This paper has focused on the maximal entropy change by scale to estimate the size of image structures. The minimal change by scale, however, indicates especialy stable scales with respect to evolution time. We expect these scales to be good candidates for stopping times in nonlinear diffusion scale-spaces.
- The entropies in this paper are global measures. For topics such as focus-of-attention it would be interesting to study local variants of them.

It should be emphasized that the analysis carried out in this paper is directly transferable to the analysis of multifractals, gray-value moments, and gray-value histograms.

7.6 Acknowledgments

This research has been funded in parts by the Real World Computing Partnership, the Danish National Research Council, and the EU-TMR

Project VIRGO. We thank Peter Johansen, Mads Nielsen, Luc Florack, Ole Fogh Olsen, and Robert Maas for many discussions on this topic. Preben Graae Sørensen from the Department of Chemistry at the University of Copenhagen (Jensen et al., 1998) has supplied the spiral image in Figure 7.4. The images in Figure 7.5 are taken from the 'VisTex' collection (Picard et al., 1995).

7.7 Relations to Gray Value Moments, Histograms, and Multifractal Spectra

The gray-value moments of an image are defined as

$$m_\alpha(\boldsymbol{p}) = \sum_{i=1}^{N} p_i^\alpha. \tag{7.19}$$

From the definition of S_α in (7.7) it is clear that there is a one-to-one relation to m_α.

Let the image $(p_1, .., p_N)^T$ consist of M distinct gray values $v_1, .., v_M$ occurring $f_1, .., f_M$ times. We may use this gray-value histogram $\boldsymbol{f}$ to rewrite the moments as

$$m_\alpha(\boldsymbol{p}) := \sum_{j=1}^{M} f_j v_j^\alpha. \tag{7.20}$$

Considering the moments $m_0, \ldots, m_{M-1}$ gives the relation:

$$\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ \vdots \\ m_{M-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ v_1 & v_2 & \ldots & v_M \\ v_1^2 & v_2^2 & \ldots & v_M^2 \\ & & \ddots & \\ v_1^{M-1} & v_2^{M-1} & \ldots & v_M^{M-1} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_M \end{bmatrix}$$

The system matrix is a so-called Vandermonde matrix. By induction over M the determinant can be shown to be $\prod_{1 \le n < m \le M} (v_m - v_n)$.

Since $v_j, j = 1, \ldots, M$ are distinct, the matrix is invertible (but ill-conditioned). Thus there is a one-to-one relation between the moments $m_0, \ldots, m_{M-1}$ and the histogram $f_1, \ldots, f_M$.

The equivalence of the multifractal spectrum and the generalized entropies is discussed in (Halsey et al., 1986; Véhel and Vojak, 1998).

Chapter 8

Some Theorems on Continuous Histograms

This chapter discusses continuous histograms of one dimensional functions. We define a continuous histogram by the first order structure of the function, and as such they seem to be one-to-one mappings of the function up to translation and mirroring. This is proven for a large class of functions including almost all uneven polynomials. We further show that if the function has an extremum, then the histogram can be used to find the first non-zero derivative of the function in almost all cases.

8.1 Why Study Continuous Histograms?

The gray-value histogram is a simple function with a wide range of applications. For example, in signal and image processing the shape of the histogram is used to reduce the number of function values or in the extreme case to segment the signal. In coding theory the histogram is used as a basis of code design, since the histogram dictates the lengths of the optimal codes. In this article we will examine the continuous histogram as the first step in obtaining a deeper understanding of the discrete histogram.

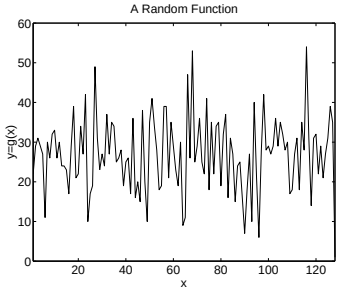


Figure 8.1: A random positive function.

We will in the following examine the central importance of the discrete histogram in various fields, and thereafter introduce the continuous histograms as the limit of infinitely finely sampled discrete histograms.

8.1.1 Some One-To-One Relations with the Discrete Histogram

We will shortly digress on the relations in the discrete setting, since it is here easiest shown that gray-value histogram is equivalent to the spectrum of gray-value moments, the generalized entropies and the multifractal spectrum. An example of these representations for the random function in Figure 8.1 is shown in Figure 8.2.

Let $\vec{g} = (g_1, \ldots, g_M)^T$ be a discrete function where $g_m > 0$ for $m = 1 \ldots M$. For each distinct function value $y_1, \ldots, y_N$ of $\vec{g}$ we may count the frequency of occurrence $\vec{h} = (h_1, \ldots, h_N)^T$. We call $\vec{h}$ the discrete gray-value histogram.

The gray-value moment of order α is defined as (Gonzales and Woods, 1993),

$$m_\alpha = \sum_{n=1}^{N} h_n y_n^\alpha, \tag{8.1}$$

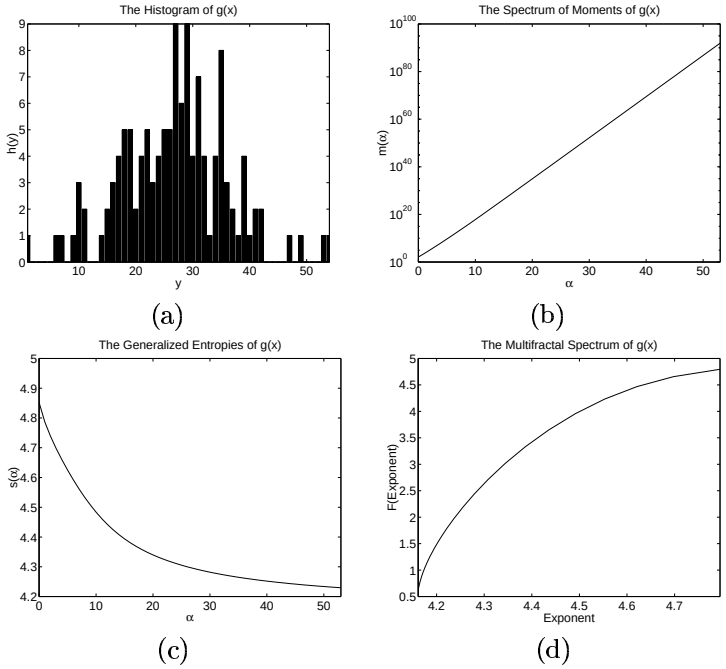


Figure 8.2: Four equivalent representations of the random function in Figure 8.1. The normalised discrete histogram is shown in (a). (b) is the spectrum of moments on logarithmic scale, (c) is the generalized entropies, and (e) is the multifractal spectrum.

and the spectrum of moments $m_0, \ldots, m_{N-1}$ gives the following relation,

$$\begin{bmatrix} m_0 \\ m_1 \\ \vdots \\ m_{N-1} \end{bmatrix} = \begin{bmatrix} y_1^0 & y_2^0 & \ldots & y_N^0 \\ y_1^1 & y_2^1 & \ldots & y_N^1 \\ & & \ddots & \\ y_1^{N-1} & y_2^{N-1} & \ldots & y_N^{N-1} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{bmatrix}.$$

The above matrix $\{y_n^\alpha\}$ is a Vandermonde matrix. The matrix can be shown to be ill-conditioned but invertible for all N. This shows that the spectrum of moments and the histogram are equivalent representations.

For a discrete function $\vec{g}$ the generalized entropy S of order α is defined as (Rényi, 1976c; Rényi, 1976a; Rényi, 1976b),

$$S_\alpha = \frac{1}{1-\alpha} \log \left(\sum_{m=1}^{M} \left(\frac{g_m}{c} \right)^\alpha \right), \tag{8.2}$$

where $c = \sum_{m=1}^{M} g_m$ is a constant. The generalized entropy is not defined for $\alpha = 1$, but the limit of α going to 1 from below or above can be seen to be $S_1 = -\sum_{m=1}^{M} (g_m/c) \log(g_m/c)$ by l'Hôpital's rule. This function is usualy taken as part of the generalized entropies.

Using $\sum_{n=1}^{N} h_n y_n^\alpha = \sum_{m=1}^{M} g_m^\alpha$ we rewrite (8.2) a

$$m_\alpha = \exp\left((1-\alpha) S_\alpha\right) c^\alpha.$$

This demonstrates that the spectrum of moments and the generalized entropies are equivalent representations.

Finaly, the one version of the multifractal spectrum is defined as the Legendre transform of $(\alpha - 1) S_\alpha$ (Halsey et al., 1986). The Legendre transform is invertible, hence the multifractal spectrum is equivalent to the generalized entropy.

8.1.2 Continuous Histograms

Continuous histograms are the limit of discrete histograms, when the sampling both of space and intensity values is infinitely small. Consider a simple third degree polynomial. What happens to the discrete

histogram, when the function is sampled at finer and finer resolution? In Figure 8.3 we see the evolution of a third degree polynomial as the number of sampling points is increased. Besides the interference between the two sampling rates causing the Moiré pattern, we see that the extrema for the polynomial give rise to pole like structures in the histogram.

In the limit of infinitely fine sampling, any analytical function will be dominated by the linear term. Let us therefore examine histograms of straight lines. Lines give rise to uniform histograms. In Figure 8.4 are two lines of different slope given. If we investigate the amount of a straight line that is projected onto an interval on the horizontal axis illustrated by the two horizontal lines, we see that a line of high slope will have a smaller projection than a line with low slope. Particularly we see that the only lines that do not give uniform histograms are lines of zero slope. Zero slope lines project everything into a single point on the vertical axis. We make the following observations regarding the projection of straight lines:

- It is independent on the particular offset of the domain.
- Mirroring of the domain yields does not change the projection.
- Two lines with the same absolute slope have identical projections

Linear structures are the basis of all analytical functions, hence the above is easily generalized to all analytical functions.

We are thus motivated to used the following definition for continuous histograms.

Definition 8.1 (Continuous Histogram).
A continuous histogram of a monotonic, one dimensional function $y = g(x)$, where $g \in \mathcal{C}^1$, is defined as,

$$h(y) = \frac{1}{\overline{|g'(x)|}}.$$

A continuous histogram of a non-monotonic function is defined as the

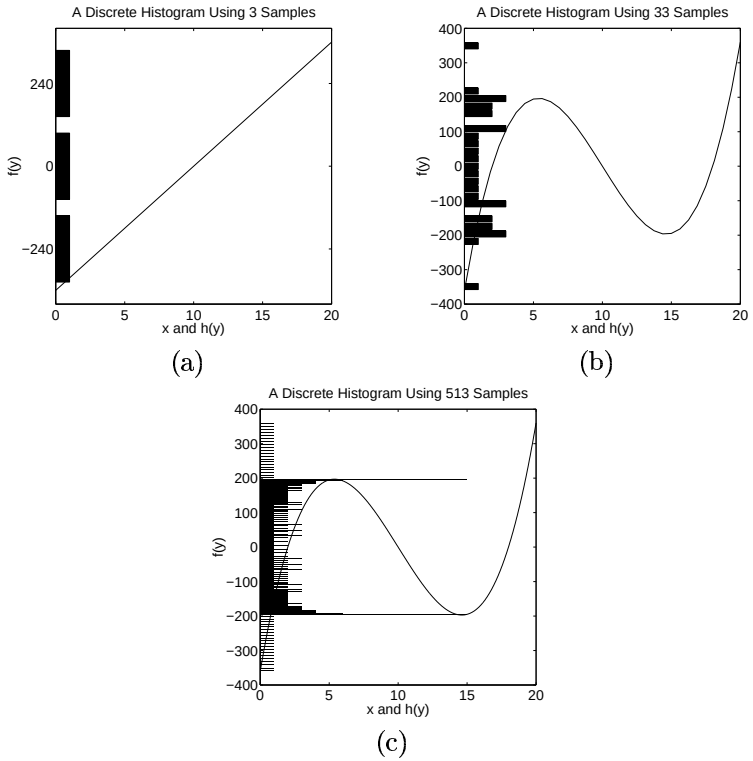
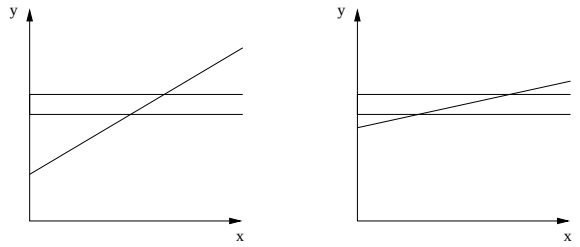


Figure 8.3: A function and its histogram under different sampling rates. The function $f(x) = (x-2)(x-10)(x-18)$ is shown using 3, 33, and 513 sampling points on the horizontal axis. The histogram is shown using the same sampling rates as a projection onto the vertical axis. The interference between these two samplings causes the Moiré patterns especialy noticeable in (c).



(a) (b)

Figure 8.4: Two lines of different slope and the density on the vertical axis. In (a) is shown a line with high slope and an interval of sampling on the vertical axis. In (b) is shown a corresponding line with low slope.

sum over intervals where the function is monotonic,

$$h(y) = \sum_{x:g(x)=y} \frac{1}{|g'(x)|}. \tag{8.3}$$

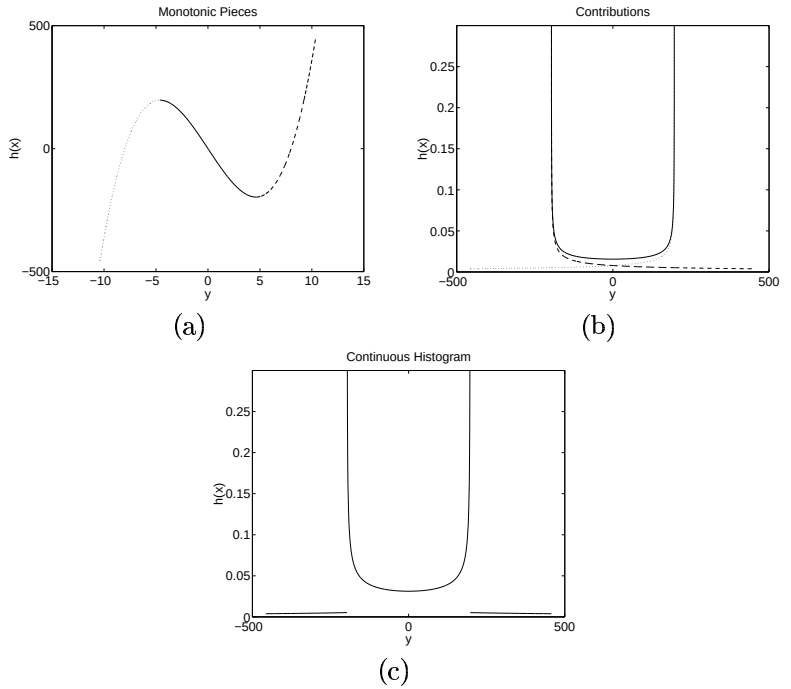
In Figure 8.5 and 8.6 are shown two examples of continuous histograms.

8.1.3 Final Introductory Remarks

The goal of this work is to show that the continuous gray-value histogram is a complete representation of a one dimensional function up to translation and mirroring.

Related to this article are reconstructions from zero-crossings in the image (Yuille and Poggio, 1983; Hummel and Moniot, 1989), reconstructions from the sign information of Fourier coefficients (Curtis et al., 1985), and reconstruction from Top-points (Johansen, 1997).

The process of studying continuous histograms of one dimensional functions is broken into several steps. Firstly, in Section 8.2, we study the histograms of monotonic functions. In the same section the results



(a) (b)

(c)

Figure 8.5: A non-monotonic function, the elements in the sum of the histogram, and the continuous histogram for the function $f(x) = x^3 - 64x$. The function is shown in (a) indicating the three monotonic pieces. In (b) is the contribution of each piece to the histogram, and in (c) is the continuous histogram shown.

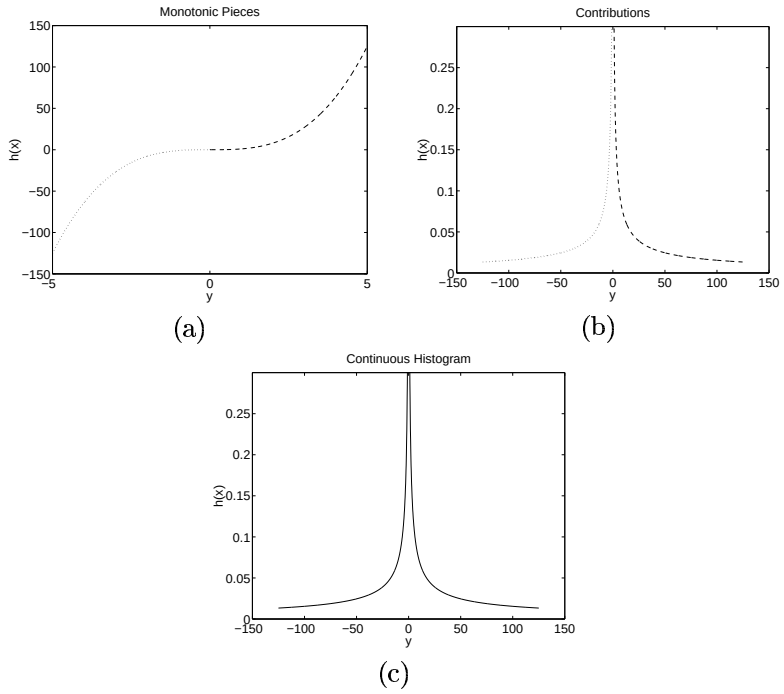


Figure 8.6: A monotonic function with a singularity, the elements in the sum of the histogram, and the continuous histogram for the function $f(x) = x^3$. The function is shown in (a) indicating the two monotonic pieces chosen. In (b) is the contribution of each piece to the histogram, and in (c) is the continuous histogram shown.

are extended to partially injective functions. We prove that continuous histograms of partialy injective functions uniquely define a function up to translation and mirroring of the domain. In the class of all polynomials, all odd degree polynomials have an injective neighbourhood, and are thus uniquely defined by their histogram up to translation and mirroring. The discussion on one dimensional functions that do not have an injective interval is split into two: Section 8.3 and 8.4. Section 8.3 will examine the algebraic structure available through the poles of the histogram, and Section 8.4 will discuss the analytical structure available in the poles. The uniqueness of the histograms for non-injective one dimensional functions is not proven, but it is conjectured to be true by the algebraic and analytical analysis performed.

8.2 Monotonic Functions

The simplest functions to reconstruct from histograms are monotonic functions. We will prove the following.

Proposition 8.1 (Histograms of Monotonic Functions). *Let g be a monotonic and analytical function. The continuous histogram of g is a full representation up to a translation and mirroring of the domain.*

Proof. Assume the continuous histogram of $g(x)$ is given by $h(y)$. Since g is monotonic and analytic we may write the spatial coordinates of g as

$$x(y) = x(y_0) \pm \int_{y_0}^{y} h(y)\, dy$$

up to the offset $x(y_0)$ and sign of g'. Since the histogram is monotonic we may write $g(\pm x + x(y_0)) = x^{-1}(y)$. The arbitrary offset $x(y_0)$ corresponds to an arbitrary translation and the sign to a mirroring of the domain. This completes the proof.

□

In Figure 8.7 is given an example of the continuous histogram of $\tanh(x)$ and its corresponding reconstruction. In Figure 8.8 are the same functions shown for a monotonic interval of $\cos(x)$. By the

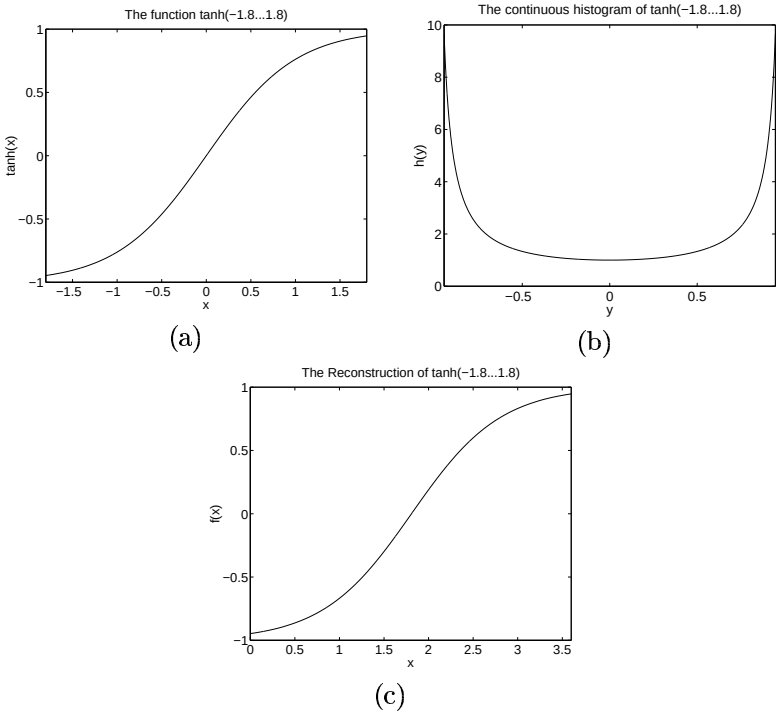


Figure 8.7: The function $\tanh(x)$ in a limited interval (a), its continuous histogram (b), and the reconstructed function (c). Note that the reconstructed function differs only by a translation of the domain.

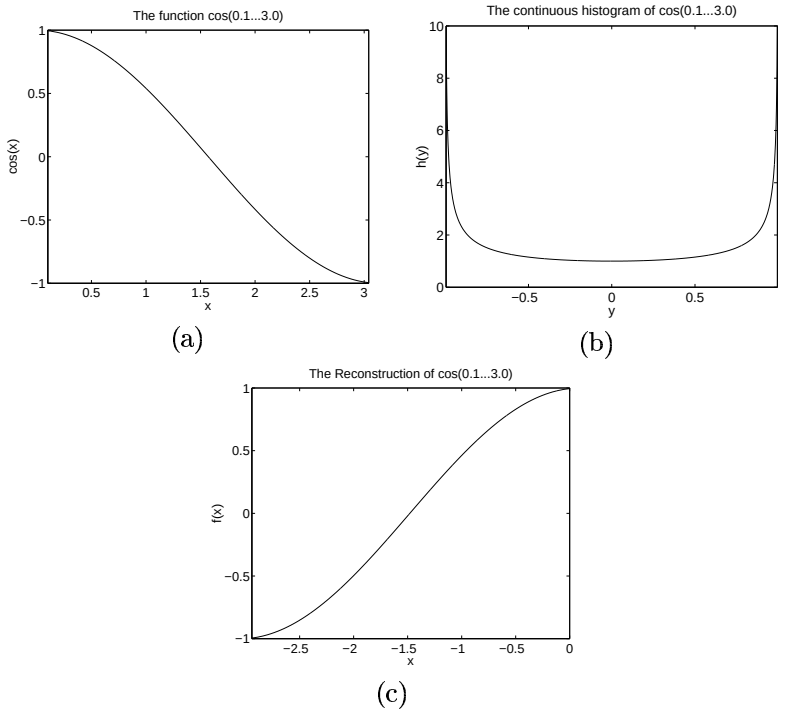


Figure 8.8: The function $\cos(x)$ in a limited interval (a), its continuous histogram (b), and the reconstructed function (c). Note that the reconstruction function differs by a translation and a mirroring of the domain.

reconstructed function from the histogram we see that the histogram represents the function up to translation and mirroring of the domain.

We will now examine a function which is partialy injective.

Definition 8.2 (Partially Injective Functions).
An analytical one dimensional function $g(x)$ is partially injective if there exists an interval Y such that if $g(x_1) = g(x_2)$ and $g(x_1) \in Y$ then $x_1 = x_2$..

For the class of partialy injective functions we can use the above proposition to prove the following.

Lemma 8.1 (Histograms of Partially Injective Functions).
Let $g(x)$ be a non-monotonic but partially injective function. The continuous histogram defines a class of functions differing from $g(x)$ only by translation and mirroring of the domain.

Proof. By Proposition 8.1 we may reconstruct g in the injective neighbourhood up to translation and mirroring. Since g is analytic, the Taylor series for the neighbourhood will converge to g. This completes the proof. □

We note that the above lemma is valid for all polynomials of odd degree. Further, for the class of polynomials, it is always possible to identify the injective intervals, since the pole structure identifies the extrema.

8.3 Algebraic Structure of Poles

Extremal points of the function will give rise to poles in a continuous histogram. We will in the following two sections show that the structure of the poles is directly linked to higher order derivatives of the function at the extrema. In this section we will examine the algebraic structure of the pole and in the next section we will examine the analytical structure of poles. As in the previous section we note that the histogram is invariant to translation and mirroring of the domain.

In Figure 8.5 and 8.6 is shown two third degree polynomials and their continuous histograms. As should be expected, we see in Figure 8.5 that for the maximum the pole is continuous from below and discontinuous from above and vice versa for the minimum. In Figure 8.6 we note that although there is no extremum, the derivative of g is zero and the histogram has a pole that is continuous both from above and below.

In a pole y_1, the sum in (8.3) is completely dominated by the single term originating from the extremum $g(x_1)$, and hence:

$$\lim_{y \to y_1} h(y)|g'(x_1)| = 1.$$

This can be used to obtain the singularity structure in the extrema. Note that the limit taken in this and the following assumes a direction, i.e. from below for a maximum and above for a minimum. These directions can be inferred directly from the continuous histogram.

Although the continuous histogram $h(y)$ is given as a function of y and not x, its structure at poles (x_k, y_k) reveals information on the spatial structure of $g(x)$ at the singularity.

Proposition 8.2 (Structure from Histogram).
Let $g(x)$ be an analytical function for which $g'(x_k) = 0$ and if $g'(x_l) = 0$ and $g(x_l) = g(x_k)$ then $x_l = x_k$. Both the multiplicity m_k and the $g^{(m_k+1)}(x_k)$ may be obtained directly from the histogram $h(y)$.

Proof. Denote the known intensity values of the poles by y_k, the corresponding unknown spatial positions of the extrema x_k, the multiplicity of each pole by m_k, and the enumerator of the corresponding partial fraction by α_{km_k}. These values can be found as follows: Obtaining y_k from the histogram is the simple process of noting the function values of the poles. In the neighbourhood of a pole the function is similar to $g(x) = y = c_k x^{m_k+1}$. Disregarding the constant we may find the multiplicity m_k using the inverse $x = |y|^{1/(m_k+1)}$. The multiplicity is thus given as the smallest positive integer n for which,

$$\lim_{y \to y_k} |y - y_k|^{n}$$

$$\overline{n+1}\, h(y) \leq \infty.$$

In practice however the limit cannot be handled correctly in a computer, and we are forced to examine the convergence to y_k.

To solve for the local structure we will translate the coordinate system and examine the extremum for $y = 0$. We will now examine the effect of the constant c_k in $g(x) = y = c_k x^{m_k+1}$. Write the histogram as,

$$\lim_{y\to y_k} h(y) = \sum_{\{x:g(x)=y\}} \frac{1}{|g'(x)|} = \frac{2^{\text{odd}(m_k)}}{\left((m_k+1)|c_k|^{\frac{1}{m_k+1}}|y|^{\frac{m_k}{m_k+1}}\right)},$$

where odd is the indicator function defined as,

$$\text{odd}(m_k) = \begin{cases} 1 & \text{if } m_k \text{ is odd} \\ 0 & \text{if } m_k \text{ even} \end{cases}$$

We see that the limit in y is easily related to c as:

$$c_k = \pm \lim_{y\to y_k} \left(\frac{2^{\text{odd}(m_k)}}{(m_k+1)h(y)|y|^{\frac{m_k}{m_k+1}}} \right)^{m_k+1}.$$

The above is easily generalized to $g(x) = y = c_k(x - x_k)^{m_k+1} + y_k$ and related to the structure through $g^{(m_k+1)}(x) = \pm(m_k+1)!c_k$. If m_k is odd, then the sign of c_k can be obtained by a simple analysis of the continuity structure of $h(y)$ as demonstrated by Figure 8.5. This completes the proof.

□

In the rest of this section we will study regular polynomials.

Definition 8.3 (Regular Polynomial).
A polynomial of degree L for which the derivative has $L-1$ real roots we call a regular polynomial.

In passing we note that all polynomials of finite degree can be made regular using the Heat Equation $\partial_t = \partial_{x^2}$ for some negative t. We will now discuss the following.

Conjecture 8.1 (Histogram of Regular Polynomials).
A one dimensional regular polynomial g can be represented by the continuous histogram up to a translation and mirroring of the x-axis.

The histogram $h(y)$ of a regular L'th degree polynomial $y = g(x)$, whose derivative has all real roots, will have K poles each with a multiplicity m_k such that $\sum_{k=1}^{K} m_k = L - 1$.

For a Regular Polynomial $g(x) = \sum_{l=0}^{L} a_l x^l$ we now have the following $2K + L - 1$ equations for $L + 1$ unknown $a_0, \ldots, a_L$,

$$\begin{aligned} g(x_k) &= y_k, \\ g'(x_k) &= \ldots = g^{(m_k)}(x_k) = 0, \\ g^{(m_k+1)}(x_k) &= \pm(m_k+1)!c_k. \end{aligned} \tag{8.4}$$

Due to translational invariance we might as well fix $x_1 = 0$. Together with the constraints on the derivatives this immediately yields the following equations,

$$\begin{aligned} g(x_1) &= a_0 = y_1 \\ a_1 &= \ldots = a_{m_1} = 0 \\ a_{m_1+1} &= \pm(m_k+1)!c_k, \end{aligned}$$

leaving $2K + L - 1 - 2 - m_1$ equations.

We see that when all roots are equal, $K = 1$ and $m_1 = L - 1$ implying zero unused equations, which proves this special instance of the conjecture. We will in the following subsections give two examples, where the conjecture is true.

8.3.1 Example: Regular Polynomial, One Extremum

For an example of the above consider the polynomial

$$g(x) = \frac{1}{6}a_3x^3 + \frac{1}{2}a_2x^2 + a_1x + a_0, \tag{8.5}$$

where $g'(x)$ has two real roots. I.e.

$$g'(x) = \frac{1}{2}a_3(x - x_1)(x - x_2)$$

for real x_1 and x_2.

A third degree regular polynomial with just one extremum has $x_1 = x_2$. Such a function and its histogram is drawn in Figure 8.6. The polynomial is monotonic and we may perform a Taylor expansion as mentioned earlier, but for the sake of illustration we will reconstruct algebraicaly. The values $\{g(x_1) = y_1;\ m_1 = 2;\ g'(x_1) = g''(x_1) = 0;\ c_1\}$ are all obtained from the histogram. Fixing $x_1 = 0$ we immediately get,

$$\begin{aligned} a_0 &= y_1, \\ a_1 &= 0, \\ a_2 &= 0, \\ a_3 &= \pm 6c_1. \end{aligned}$$

The unknown sign of a_3 is due to the undetermined mirroring.

8.3.2 Example: Regular Polynomial, Two Extrema

Let us continue with the example in (8.5) and now assume that $x_1 \neq x_2$. Such a function and its histogram is drawn in Figure 8.5. The polynomial has two extrema, and we hence create 3 monotonic pieces: $x \in \{-\infty, x_1\}$,$x \in \{x_1, x_2\}$, and $x \in \{x_2, \infty\}$, assuming position of the two extrema to be $x_1 < x_2$. The values $\{g(x_k) = y_k;\ m_k = 1;\ g'(x_k) = 0;\ c_k\}$ are all obtained from the histogram (the x_k's are unknown). We set $x_1 = 0$ and get,

$$\begin{aligned} a_0 &= y_1, \\ a_1 &= 0, \\ a_2 &= 2c_1. \end{aligned}$$

To solve for x_2 and a_3 we use

$$\begin{aligned} g(x_2) &= 1/6a_3x_2^3 + c_1x_2^2 + y_1 = y_2, && (8.6) \\ g'(x_2) &= 1/2a_3x_2^2 + 2c_1x_2 = 0, && (8.7) \\ g''(x_2) &= a_3x_2 + 2c_1 = 2c_2. && (8.8) \end{aligned}$$

Using (8.7) and (8.8) we find $c_1 = -c_2$. Using (8.6) and (8.7) we get $x_2 = \pm\sqrt{3\frac{y_2 - y_1}{c_1}}$. The coefficient a_3 may be calculated directly by (8.8).

Again we see an undetermined sign for the a_3 coefficient (equivalently x_2) corresponding to an undetermined mirroring of the domain.

8.4 Analytical Structure of Poles

One feature of non-partialy injective functions is that they have at least one global extremum. If only one global extremum is present, then the continuous histogram degenerates to being a sum of only two terms. We will in this section try to take advantage of this. We have not been able to prove uniqueness in the sense of previous sections, but we will sketch an algorithm that has been implemented and appears to converge to the correct solution up to translation and mirroring of the domain.

Let $g(x)$ be an analytical function with one global extremum (x_k, y_k) with multiplicity m_k. Without loss of generality we will assume that this is a global minimum. For a sufficiently small constant δ the continuous histogram h is given by:

$$h(y_k + \delta) = \frac{1}{\frac{1}{g'(x_k + \epsilon)} - \frac{1}{g'(x_k - \eta)}},$$

where the constants δ, ϵ, and η are related through

$$g(x_k + \epsilon) = g(x_k - \eta) = y_k + \delta.$$

We now set $x_k = 0$ and analyse the histogram sufficiently close to y_k such that only lower order terms of g are detectable. More precisely, choose a δ such that the structure of g is sufficiently represented by the following truncated Taylor series:

$$g_L(x) = \sum_{l=0}^{L} \frac{a_l}{l!} x^l.$$

In the neighbourhood around y_k we may solve for the left and right solution of the inverse function of g_L, obtaining approximations of $\epsilon_L(\delta) \simeq \epsilon$ and $\eta_L(\delta) \simeq \eta$. Analytical solutions of these are rather complicated and we will suffice with stating that only these two solutions exist for sufficiently small δ, and that these can at least be found numericaly. We can thus write an approximation to the histogram based on g_L:

$$h_L(y_k + \delta) = \frac{1}{g'_L(\epsilon_L)} - \frac{1}{g'_L(-\eta_L)}.$$

Let us assume that the structure is known up to $L-1$, hence that a_L is the highest dominating term and unknown. For a given a_L we may solve for $\epsilon_L(\delta)$ and $\eta_L(\delta)$, hence we may write the problem as a minimisation of

$$E(a_L) = \left(h_L(y_k + \delta) - h(y_k + \delta)\right)^2,$$

and seek the solution by gradient descent. We have not proven convergence, but the experiments we have performed indicate that at least for small polynomials, the above brute force method converges to the right solution.

8.5 Discussion

In this article we have examined the structure of continuous histograms of one dimensional functions. We have proven that the histogram of partialy injective functions defines a class of functions differing only by a translation and mirroring of the domain. Further, we have examined the strong algebraic and analytical constraints on the possible function class determined by continuous histograms in general.

The contributions of this work are both to provide theoretical insight into continuous histograms, and to indicate the viability of using histograms for shape representation. For example, in the context of two dimensional shape it is known that any two dimensional contour can be represented by its curvature function up to a rotation and translation. The curvature function of a closed contour has an arbitrary phase, but by examining shape from the histogram of the curvature function, we will be able to disregard the phase completely. The histogram is further invariant to mirroring, which in terms of the curvature function corresponds to a mirroring in the two dimensional plane. Such invariances are often desirable for some shape recognition tasks.

8.6 Acknowledgments

This work was initiated by discussions with Robert Maas in Utrecht during the Scale-Space '97 Conference, where we discussed the possibility of reconstruction from histograms for a large number of kernels. The present weaker result using continuous histograms was spawned by later discussions with Mads Nielsen, Peter Johansen, Ole Fogh Olsen, Jørgen Sand, and Joachim Weickert.

Chapter 9

On the Invariance of Saliency Based Pruning Algorithms[1]

9.1 Introduction

For some function classes such as the feed forward neural networks, the number of parameters is very large when compared to the usual size of datasets to be fitted. To give an example, in the simplest universal feed forward network $f : \mathbb{R}^N \to \mathbb{R}^M$ (Hornik, 1989; Cybenko, 1989), the number of parameters grow as $d(M+N+1)$, where d is the number of internal nodes (or hidden neurons).

To reduce complexity and increase generalization, a function class can be analyzed by examining each individual parameter for its importance or saliency. The process of removing parameters based on saliencies is known as pruning and is the subject of this chapter. We will illustrate how a specific pruning scheme, Optimal Brain Damage

[1] An earlier version of this chapter has been published as a technical report (Sporring, 1997). The current version is submitted for journal publication.

(OBD) (Cun et al., 1990), can be used to generate a similarity class of algorithms based on invariance, which in turn can be interpreted in a statistical manner as a Maximum A Posteriori (MAP) or information theoretical code length functional, and it is thus shown how OBD can be interpreted in terms of the implicit prior on the function clas, usualy the feed forward networks.

Before we begin, the reader should note that although the foundations of MAP and coding are very different, there is in the idealized code length setting applied here, a one to one correspondence between the two. Idealized code lengths are determined through Shannon's entropy-inequality (Rissanen, 1989), a

$$L(\theta) = -\log P(\theta),$$

where L is the code length for the particular entity θ and P is its corresponding probability. Under the assumption that P is known, there exists algorithms, such as the Huffman and especialy the Arithmetic coding algorithm, that approach an equality of the above. Conversely, it is straightforward through the equality to design a probability distribution given a set of complete prefix codes. We are thus in this loose sense free to choose the formalism best suited for our needs.

9.2 Pruning

Fitting a function to a set of data points is often accomplished by minimizing an error function $E(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the set of parameters. The definition of saliency as we use it in this chapter is the increase in E when one or more parameters are removed, i.e. set to zero. The increase by removal of the parameter set $\{\theta_{i_1}, \ldots, \theta_{i_n}\}$ will be called $\Delta_{\{\theta_{i_1},\ldots,\theta_{i_n}\}}E$, and an ordering is thus induced,

$$\Delta_{p_{i-1}}E \geq \Delta_{p_i}E \geq \Delta_{p_{i+1}}E$$

where we used the sloppy notation of p_j to denote a set of parameters. The exact pruning decision performed is not of importance to the work presented in this chapter, as long as the decision is based only on the

ordering. Generaly the set of parameter removals that generate the lowest increase in the error function is pruned.

The exact increase is often too computationally expensive to evaluate, and for analytical error functions (usualy implying analytical functions) the ordering may be estimated by a truncated Taylor serie

$$\begin{aligned}\Delta E(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) &\equiv E(\boldsymbol{\theta} - \Delta\boldsymbol{\theta}) - E(\boldsymbol{\theta}) \\ &= -\sum_i \frac{\partial E(\boldsymbol{\theta})}{\partial \theta_i}\Delta\theta_i + \frac{1}{2}\sum_i\sum_j \frac{\partial^2 E(\boldsymbol{\theta})}{\partial\theta_i\partial\theta_j}\Delta\theta_i\Delta\theta_j - \dots.\end{aligned}$$

As an example, the error functional used in OBD is $E = \sum_n (y_n - f(x_n))^2$ for the data points $\{x_n, y_n\}$ and the function f, and the Taylor series for ΔE is truncated to second order.

A mathematicaly as well as computationaly convenient restriction is to consider only single parameter prunings. This reduces the number of saliencies to be computed to equal the number of parameters (not yet pruned), and it simplifies the Taylor series to

$$\Delta_p E(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) = -\theta_p \frac{\partial E(\boldsymbol{\theta})}{\partial \theta_p} + \theta_p^2 \frac{1}{2}\frac{\partial^2 E(\boldsymbol{\theta})}{\partial \theta_p^2} - \dots,$$

for each parameter θ_p. Note that in this case, $\Delta\boldsymbol{\theta} = [0, .., 0, \theta_p, 0, .., 0]^T$.

How well the truncated Taylor series approximates ΔE is usualy ignored in the literature. Further, the ordering itself does not indicate to what extend the pruning is to be continued. This must be determinedby exterior constraints such as generalization maximization, see e.g. (Sporring, 1995; Svarer et al., 1993; Rasmussen, 1993) and the references therein and many others.

9.3 Monotonic Transformations of Pruning Order

For the simplicity of the following argument we will investigate single parameter pruning algorithms, but note that the results holds for multi

parameter prunings as well. Assume that we have an ordering of the parameters such that,

$$\Delta_{p_{i-1}} E \geq \Delta_{p_i} E \geq \Delta_{p_{i+1}} E.$$

It is at once noticed that since monotonic transformation with positive slope preserve inequality, the above ordering is also unaffected,

$$T(\Delta_{p_{i-1}} E) \geq T(\Delta_{p_i} E) \geq T(\Delta_{p_{i+1}} E),$$

I.e. continuous transformation $T : \mathbb{R} \to \mathbb{R}$ with $\partial_x T \geq 0$ for all x does not affect the pruning order. We will now study a linear transformation $T(\Delta E) = a\Delta E + \mathrm{b}$, for constants $a > 0$ and b, and show that the pruning algorithms described in this chapter can be interpreted in terms of a model expectancy.

Examine the following function,

$$L(\boldsymbol{\theta}) = \alpha E(\boldsymbol{\theta}) + \sum_i \beta_i \log|\theta_i| + \gamma, \tag{9.1}$$

where α, β_i, and γ are constants. α must be greater than or equal zero, and the set of β_i's must be chosen such that the saliency order is not disturbed. Generaly we will assume that $\beta_i = 0$ when $\theta_i = 0$ and use the convention that $0 \log 0 = 0$.

Proposition 9.1 (Existence). *For $\alpha > 0$ and a constrained set of β_i's L preserves the pruning order of any analytical error function E in a Taylor series truncated to finite order.*

Proof. The proof is given in Section 9.7. □

Proposition 9.2 (Uniqueness). *For $\alpha > 0$ and a constrained set of β_i's L is the only functional of any analytical error function E for which the change of L is a linear function of the change of E in the Taylor series truncated to finite order.*

Proof. See Section 9.8 for the proof. □

There are several key points to notice. First of all, the particular set of β_i's where $\beta_j = \beta$ for all j does not upset the pruning order. To see this, write the constraints on β_i as (Equation 9.3),

$$\alpha\frac{\Delta_{p_{i-1}}E - \Delta_{p_i}E}{\sum_{j=1}^{J} j^{-1}} - \beta_{i-1} \geq -\beta_i \geq \alpha\frac{\Delta_{p_{i+1}}E - \Delta_{p_i}E}{\sum_{j=1}^{J} j^{-1}} - \beta_{i+1}$$

where J is the truncation order. For identical β_j's the original order is retained.

Secondy, this particular choice of identical constants β_j's is precisely the limit for the truncation order going towards infinity, since the sum in the denominator will tend to infinity as J does, hence the band of different allowable β_j's will tend to zero, i.e. $\beta_j \to \beta$ for all j as $J \to \infty$.

Finaly, if E is an analytical function then L is, too. We have a semi-group property in the sense that we can define two sequential non-pruning disturbing extensions as in Equation 9.1 and get a third non-disturbing pruning. Thus defines L' a

$$L'(\boldsymbol{\theta}) = \alpha' L(\boldsymbol{\theta}) + \sum_i \beta_i' \log|\theta_i| + \gamma',$$

with a new set of constants chosen as prescribed previously, but this time based on L instead of E. This is of course just

$$L'(\boldsymbol{\theta}) = \alpha'\alpha E(\boldsymbol{\theta}) + \sum_i (\alpha'\beta_i + \beta_i')\log|\theta_i| + \alpha'\gamma + \gamma',$$

Again we see that the requirements to be fulfilled are

$$\alpha'\alpha\frac{\Delta_{p_{i-1}}E - \Delta_{p_i}E}{\sum_{j=1}^{J} j^{-1}} - \alpha'\beta_{i-1} - \beta_{i-1}' \geq -\alpha'\beta_i - \beta_i'$$
$$\geq \alpha'\alpha\frac{\Delta_{p_{i+1}}E - \Delta_{p_i}E}{\sum_{j=1}^{J} j^{-1}} - \alpha'\beta_{i+1} - \beta_{i+1}$$

and for $\beta_j = \beta$ and $\beta_j' = \beta'$ this requirement is trivialy fulfilled. Note that this is a different approach than choosing two different sets of β_i's both chosen from the same analytical function and then combined. This last approach is in general not a pruning order invariant.

9.4 A Prior of Saliency Based Pruning Algorithms

We will now examine the choice of $\beta_j = 1$ for all j. Equation 9.1 can be interpreted in the coding setting as the sum of code lengths of the noise model and the parameter model, and in the MAP setting as minus the logarithm of the noise probability times the prior,

$$L = L(\mathcal{D}|\boldsymbol{\theta}) + L(\boldsymbol{\theta}) = -\log P(\mathcal{D}|\boldsymbol{\theta}) - \log P(\boldsymbol{\theta}), \qquad (9.2)$$

where,

$$P(\mathcal{D}|\boldsymbol{\theta}) = \exp(-\alpha E(\boldsymbol{\theta}) - \gamma_0).$$

In the example of OBD, E is the sum over data points of the square of an $\mathcal{L}_2$ norm, and this can be interpreted as a normal product distribution with a unit standard deviation, and

$$P(\boldsymbol{\theta}) = \exp(\gamma_1)\prod_i |\theta_i|^{-\beta_i}$$
$$\simeq \exp(\sum_i -\log\eta - \log\lfloor|\delta\theta_i|\rfloor - \log\log\lfloor|\delta\theta_i|\rfloor - \ldots),$$

where $\gamma = \gamma_0 + \gamma_1$, η is a normalization constant, δ is the discretization constant to truncating reals into integers, and $\lfloor\cdot\rfloor$ is the truncation operator. The sum is continued just until the repeated logarithm yields a negative number. This last equation is also known as Rissanen's Universal Distribution of Integers (Rissanen, 1989) and most clearly demonstrates the difference between coding and the MAP methods. While the MAP methodology is best suited for continuous distributions, such as Jeffrey's semi-prior $\gamma_1/|\theta_i|$ (Jaynes, 1968), the problems of normalization and discretization is much better handled in the coding methodology. The key difference between the two is that while Jeffrey's prior can only be implemented on a finite interval of the real axis in order for it to be normalized, Rissanen's distribution is normalizable for all countable sets like the set of all positive integers. Hence using Jeffrey's prior one is concerned with the interval size D in order to evaluate the normalization constant $\gamma_1 = \int_1^D 1/x\ dx$, while one's

concern when using Rissanen's distribution is the discretization constant δ, i.e. the number of digits accounted for. It should be noted that there are of course other more sophisticated MAP and coding implementations of distributions for real numbers. These and other implementation issues concerning this coding prior can be found in (Sporring, 1995).

We may thus view the OBD pruning algorithm as a greedy algorithm searching for the minimum in Equation 9.2 by removing the least significant parameter through the error estimate. This will increase the actual error, but decrease the cost of the model.

9.5 Conclusion

This paper has demonstrated that a large class of saliency based pruning methods, where the saliency is calculated from analytical functions, can be used to generate a similarity class of pruning algorithms all having same pruning order. The (in a sense most) general extension in this similarity class is used to interpret OBD in terms of Bayesian Maximum A Posteriori (MAP) or code-length functionals and a Prior has thus been made explicit. This is found to be Jeffrey's Prior (Jaynes, 1968), which is a very natural un-committed result for the following reasons:

- Jeffrey's Prior is scale invariant in the sense that it assign equal probability mass to the intervals $1-10$, $10-100$, etc.. It is also the basis of what is known as Benford's law, which although surprising has been empiricaly validated on numerous datasets of very different nature, see e.g. (Buck et al., 1993).
- A very close relative, Rissanen's Universal Distribution of Integers is frequently used in the coding industry, and one can show (Rissanen, 1989) that it is an optimal code for large integers.

Finaly we will conclude that although OBD uses poor estimates when the parameter values of the net are large, it is a good uncommitted choice in the view of the scale invariant properties of the implicit prior.

9.6 Acknowledgments

I would especially like to thank Peter Johansen, Mads Nielsen, Luc Florack, Robert Maas and Joachim Weickert for the many and enlightening discussions during this work.

9.7 Proof of Proposition 9.1

We will now prove that the change of L (Equation 9.1) under certain restriction generates the same pruning order as the change of any analytical function E up to any but finite truncation order in the Taylor series.

The change of L can be written as,

$$\begin{aligned}\Delta L(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) &\equiv L(\boldsymbol{\theta} - \Delta\boldsymbol{\theta}) - L(\boldsymbol{\theta}) \\ &= -\sum_i \frac{\partial L(\boldsymbol{\theta})}{\partial(\theta_i)}\Delta\theta_i + \frac{1}{2}\sum_{i,j}\frac{\partial^2 L(\boldsymbol{\theta})}{\partial(\theta_i)\partial(\theta_j)}\Delta\theta_i\Delta\theta_j - \ldots\end{aligned}$$

Clearly, the mixed derivatives of the sum of the logarithms are zero, so we need only examine non-mixed terms. First we need to evaluate the n'th derivative of $\log|x|$. For simplicity write $\log|x|$ as $1/2\log x^2$, we will now prove by induction that

$$\frac{\partial^n}{\partial x^n}\frac{1}{2}\log x^2 = (-1)^{n-1}(n-1)!x^{-n}.$$

Assume that the n'th derivative is given as above. The $n+1$'th derivative is then $\frac{\partial}{\partial x}(-1)^{n-1}(n-1)!x^{-n} = (-1)^{n-1}(n-1)!(-n)x^{-n-1} = (-1)^n n!x^{-(n+1)}$. For $n=1$, the first derivative is seen to be: $\frac{\partial}{\partial x}\frac{1}{2}\log x^2 = \frac{1}{x} = (-1)^0 0!x^{-1}$, thus completing the proof.

The j'th term in the Taylor expansion of L is given as,

$$(-1)^j\frac{\partial^j L(\boldsymbol{\theta})}{j!\,(\partial(\theta_p))^j}(\Delta\theta_p)^j = (-1)^j\alpha\frac{\partial^j E(\boldsymbol{\theta})}{j!\,(\partial(\theta_p))^j}(\Delta\theta_p)^j - \frac{\beta_p}{j\theta_p^j}(\Delta\theta_p)^j.$$

We identify the first term to be α times the identical term in the Taylor expansion of E, and further because of the symmetry, i.e. $\Delta\theta_p = \theta_p$,

we quickly find that

$$\Delta_p L(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) = \alpha \Delta_p E(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) - \beta_p \sum_{j=1}^{J} j^{-1},$$

up to any finite truncation order J. The β_j's are to be chosen such that the pruning order is maintained, i.e. since $\Delta_{p_{i-1}} E \geq \Delta_{p_i} E \geq \Delta_{p_{i+1}} E$ then so must $\Delta_j L$ and thus for positive α,

$$\alpha \frac{\Delta_{p_{i-1}} E - \Delta_{p_i} E}{\sum_{j=1}^{J} j^{-1}} - \beta_{i-1} \geq -\beta_i \geq \alpha \frac{\Delta_{p_{i+1}} E - \Delta_{p_i} E}{\sum_{j=1}^{J} j^{-1}} - \beta_{i+1}. \quad (9.3)$$

This completes the proof.

9.8 Proof of Proposition 9.2

We will show that L of Equation 9.1 is the unique function that generates linear invariance to the change of any analytical function E.

A linear transformation of the change in error E must have the form,

$$\Delta L(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) = a \Delta E(\boldsymbol{\theta}, \Delta\boldsymbol{\theta}) + b,$$

where a and b are constants. We will now investigate the possible functions in the Taylor description for a and b.

The constant a is a scaling constant and it is trivially seen that if a is a function of $\boldsymbol{\theta}$ and $\boldsymbol{\Delta\theta}$ then the contribution can be eliminated by an opposite term in b. We will thus assume a to be a positive constant. The constant b is another matter. We are faced with the choice of a function h such that

$$L(\boldsymbol{\theta}) = aE(\boldsymbol{\theta}) + h(\boldsymbol{\theta}) + c$$

which in the Taylor series behaves such that

$$b = \sum_{j=1}^{J} (-1)^j \frac{\partial^j h(\boldsymbol{\theta})}{j! \, (\partial\theta_p)^j} (\theta_p)^j$$

is a constant for arbitrary but finite J. The first order terms constraint the problems to sums of functions of only one parameter. Thus either h is independent on θ_p or,

$$\frac{\partial^j h(\boldsymbol{\theta})}{(\partial\theta_p)^j} = b_p \frac{1}{(\theta_p)^j}$$

for any j and p, and constants b_p restricted as discussed in Section 9.7. Thus

$$h(\boldsymbol{\theta}) = \sum_i b_i \log|\theta_i| + b_0$$

is the only solution for arbitrary constant b_0. This completes the proof.

Chapter 10

Measuring and Modelling Image Structure: Summary

One basic problem in image processing is that of scale: Objects in images do not have predefined sizes. A general purpose image processing algorithm should thus be adaptable to handle objects of a range of sizes. A second basic problem is that of resolution: The resolution and pixel configuration of an image is most often set by the physics of the imaging device. A general purpose image processing algorithm should hence take into account the arbitrariness of the resolution and the configuration of the pixel grid.

Linear scale-space is a very useful tool to handle the above mentioned problems, as illustrated by Chapters 3 and 5 of this thesis. Linear scale-space is not the only candidate, but it is the only one that is linear (through the convolution operator), and it seems that it is the easiest to apply in image analysis. For example, differential geometric operators are very general and useful tools for image processing, and it is easy to design algorithms using such operators by linear scale-space. Unfortunately, linear scale-space dislocates features in images, and this

forces one to consider the scaling behaviour and the catastrophe structure of the feature under consideration. An example of this is given in Chapter 4. Another approach is to design scale-spaces which reduce the amount of dislocation. These are all non-linear scale-spaces, and one is studied in Chapter 7.

Scale-spaces are often considered pre-processing algorithms preparing the input for a modelling phase. This thesis has demonstrated several models. In Chapter 3 was a Hough Transform of spirals and target patterns designed for the task of tracking a dynamical chemical system, and in Chapter 5 we designed an almost one dimensional contour representation for the coding of black and white blobs. These two examples both use models of the image data, but they differ in one important fact: The model for the spirals and target patterns was designed to be a feature for human inspection, while the contour representation was designed to choose a single and complete representation of the image.

Analysing dataset via a model implies that the dataset is bisected into a model and a residual or noise part. The justification for such a perspective is that datasets obtained from physical sources are always a mixture of something deterministic and stochastic. For example even the best imaging technique will contain electric noise and discretization effects. The balance between model and noise can only be learned by example, and in the general case there does not seem to be any justification for attributing more importance to either of the model or the noise. In the general case we are thus forced to examine a model selection scheme that explicitly does not favour one over the other. The minimum description length scheme uses compression terminology to choose models, hence enforcing a common representation of the model and the noise, and thus enforcing a proper balancing of the two. This perspective is examined in the practical setting in Chapter 5 and in the theoretical setting in Chapter 9.

The basis of information theory is the entropy function, which measures the uncertainty of a stochastic source by its distribution. The concept of uncertainty is very important and has a widespread use. We have in this thesis worked with a generalization of the entropy called the generalized entropies, which is a family of uncertainty measures,

and which are all functions of the distribution of the stochastic source. One fundamental property of the entropies is that they do not include information on the spatial relations between points in the distribution. Conversely, scale-spaces on a distribution perform information reducing operation using the local spatial structure of the distribution. We have thus been led in Chapter 7 to study the interplay between entropies and scale-spaces. One major result has been that the generalized entropies are monotonic in scale, and hence may be used as a measure of causality in the sense of Koenderink (Koenderink, 1984). Although the entropies in scale-space seem simple, they also seem to be applicable in analysing spatial structures of distributions (such as images). We have examined this for the linear scale-space through the intuition that given a point of scale, a small change will tend to have the largest effect on objects of sizes comparable to the particular scale. While we do not have any analytical justification for the intuition, we have illustrated it for several examples, and relates the results to sizes of objects in the images.

A possible property of the scale-space extension of the generalized entropies is that they might describe the function or distribution up to a simple group of actions. This was the starting point for the analysis of histograms. It very quickly became apparent, that the evolution of the discrete histograms was rather complicated and we sufficed with a study of continuous histogram without a scale parameter. For continuous histograms of one dimensional functions we succeeded in Chapter 8 in proving that for a large class of functions, called partially injective, the histogram indeed describes the function completely up to translation and mirroring. Non partially injective functions were then shown to have strong bindings through the histogram, but we were not able to prove the identical result for this class of functions. It is the opinion of this author, that there still remains much to be said in this context, but that the methods used to analyse the continuous histogram might not easily be translated to analysis of discrete histograms.

Finally, in Chapter 9 we use information theoretic arguments to study a well-known model selection algorithm. Although this is not directly related to image processing, the intent has been to demonstrate that many algorithms can be interpreted in a modelling perspective.

Thus even if we build an image processing algorithm that does not make use of the minimum description length or maximum a posteriori formalism, we may often attribute a prior to the model class anyway, and we have hence implicitly used one.

Appendix A

Some Open Problems

This thesis does in no way cover all aspects of scale-space and information theory. We have however along the way stumbled onto a number of seemingly important open questions, which we either have judged as being too hard to solve for us, or to be outside the time scope of the thesis. This list is intended to suggest future fields of study, and I've mainly included it in the thesis for my own enjoyment. The questions that I have encountered frequently in the past three years are:

Curvature functions generating closed contours
For Chapter 5 we initially analysed the space of curvature functions, since these capture essential parts of shape. For coding purposes these have one major advantage compared to the implementation presented and the works of others: Curvature functions of contours in two dimensional images are one dimensional functions. This is an advantage since as a general rule we may assume that one dimensional functions have shorter description length than two dimensional functions describing the same object. However, the curvature functions we are interested in are only those describing closed contours, and for coding purposes it is therefore important to be able to distinguish these from all curvature functions. It is for instance not at all likely, that a spline approximation of a curvature function corresponds to a closed contour. Even such fundamental questions as the enumerability and measure of this subset of curvature functions we have not been able to establish.

Minimum description length and scale-space
Minimum description length or its cousin maximum a posteriori, have been used with success for model selection. It is, however, not clear, how the simplification properties of scale-space is to be incorporated into the formalisms. A desirable analysis would be the interplay between the complexity of the chosen model and the scale of the data. Consider the model class of truncated Taylor series. By the work of Nielsen (Nielsen, 1995) we know that scale-space damps terms exponentially by order. I.e. smoothing a dataset is equivalent to minimizing,

$$E = \sum_x \left((f(x) - g(x))^2 + \sum_j \frac{t^j}{j!} \frac{\partial^j f(a)}{\partial x^j} \right),$$

where $f(x)$ is the resulting function, g is the original, and t is the scale. It thus seems plausible that the complexity of the model class should decrease exponentially with scale. Such a result is quickly established in the Fourier representation of periodic signals, but it is not clear how to generalize this to other function classes.

Analytical verification of scale selection paradigm
In Chapter 7 we used the point of maximal entropy change by logarithmic scale as a scale selection paradigm. It is however unsatisfying that this paradigm is not verified analytically even for simple functions. The main problem being the sum under the logarithm. This seems to be a hard problem.

Choice of local entropies
In Chapter 7 we studied the global behaviour of generalized entropies and their use in image processing. Carrying these methods to local neighbourhoods is desirable for example in segmentation by texture tasks. However, there exist several ways lo-

cal entropies may be defined, and it is not clear which is best. Theoretically, the problem is the monotonic behaviour of the generalized entropies in linear scale-space. Scale-space is an implementation of the physical process of heat diffusion. By the second law of thermodynamics, at least the entropy of the total distribution will be monotonic. This is definitely not the case locally. Hence it is not known if the change of entropies is the best function to study. Alternatively, we may define a separate scale-space for each local neighbourhood by extracting possibly overlapping sub-distributions, and apply the scale-space to this. Then will the monotonicity properties hold for each local neighbourhood, and we may do local size estimations and hence texture segmentation etc.. The overall choice may depend on the task to solve.

Non partially injective functions

In terms of the effort we have put into the analysis of continuous histograms it is clear that we consider it important to prove the same result for non partially injective functions as we have for partially injective functions.

We thus end this thesis with a list of what we feel are interesting open questions connected to the work presented.

Bibliography

Alvarez, L., Guichard, F., Lions, P.-L., and Morel, J.-M. (1993). Axioms and fundamental equations of image processing. *Archive Rational Mechanics and Analysis*, 123(3):19–257.

Alvarez, L. and Morel, J.-M. (1994). Morphological approach to multiscale analysis: From principle to equations. In (Haar Romeny, 1994), pages 229–254.

Asada, H. and Brady, M. (1986). The curvature primal sketch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(1):2–14.

Banerjee, S., Niblack, W., and Flickner, M. (1996). A minimum description length polygonal approximation method. Technical Report RJ 10007 (89096), IBM, Almaden Research Center, 650 Harry Road, San Jose, CA 95120-6099, USA.

Baxter, R. A. and Oliver, J. J. (1994). MDL and MML: Similarities and differences. Technical Report TR207, Dept. of Computer Science, Monash University, Clayton 3168, Australia.

Blom, J. (1992). *Topological and Geometrical Aspects of Image Structure*. PhD thesis, University of Utrecht, Department of Medical and Physiological Physics, Utrecht, The Netherlands.

Blom, J., Haar Romeny, B. M. t., Bel, A., and Koenderink, J. J. (1993). Spatial derivatives and the propagation of noise in Gaussian scale-space. *Journal of Visual Communication and Image Representation*, 4(1):1–13.

Boomgaard, R. v. d., Dorst, L., Schavemaker, J., and Smeulders, A. W. M. (1996). Quadratic structuring functions in mathematical morphology. In *Mathematical Morphology and its applications to signal processing 3*.

Boor, C. d. (1978). *A Practical Guide to Splines*. Springer-Verlag.

Brink, A. D. (1996). Using spatial information as an aid to maximum entropy image threshold selection. *Pattern Recognition Letters*, 17(1):29–36.

Brink, A. D. and Pendock, N. E. (1996). Minimum cross-entropy threshold selection. *Pattern Recognition*, 29(1):179–188.

Buck, B., Merchant, A. C., and Perez, S. M. (1993). An illustration of Benford's first digit law using alpha decay half lives. *European Journal of Physics*, 14:59–63.

Chaitin, G. J. (1966). On the length of programs for computing finite binary sequences. *Journal of the ACM*, 13:547–569.

Charbonnier, P., Blanc-Féraud, L., Aubert, G., and Barlaud, M. (1994). Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of the 1st International Conference on Image Processing*, volume 2, pages 168–172, Austin, Texas, USA. IEEE Computer Society Press.

Chaudhuri, B. B. and Sarkar, N. (1995). Texture segmentation using fractal dimension. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):72–77.

Chen, M.-H. and Chin, R. T. (1993). Partial smoothing splines for noisy +boundaries with corners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1208–1216.

Clarke, B. S. and Barron, A. R. (1990). Information-theoretic asymptotics of bayes methods. *IEEE Transactions on Information Theory*, 36(3):453–471.

Cramér, H. (1946). *Mathematical Methods of Statistics*. Princeton University Press.

Cros, M. C. and Hohenberg, P. C. (1993). Pattern formation outside equilibrium. *Reviews of Modern Physics*, 65:854–865.

Cun, Y. L., Denker, J., and Solla, S. (1990). Optimal brain damage. In Touretzky, D., editor, *Advances in Neural Information Processing Systems*, pages 598–605. San Mateo.

Curtis, S. R., Oppenheim, A. V., and Lim, J. S. (1985). Signal reconstruction from fourier transform sign information. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-33(3):643–657.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:303–314.

Damon, J. (1997). Local Morse theory for Gaussian blurred functions. In (Sporring et al., 1997), chapter 11, pages 147–163.

Dom, B. E. (1996). MDL estimation for small sample sizes and its application to linear regression. Technical Report RJ 10030 (90526), IBM, Almaden Research Center, 650 Harry Road, San Jose, CA 95120-6099, USA.

Elias, P. (1975). Universal codeword sets and representations of the integers. *IEEE Transactions on Information Theory*, IT-21(2):194–203.

Field, R. J. and Burger, M. (1985). *Oscillations and Traveling Waves in Chemical Systems*. John Wiley & Sons, New York.

Florack, L. (1997). *Image Structure*. Computational Imaging and Vision. Kluwer Academic Publishes, Dordrecht.

Florack, L. M. J., Haar Romeny, B. M. t., Koenderink, J. J., and Viergever, M. A. (1992). Scale and the differential structure of images. *Image and Vision Computing*, 10(6):376–388.

Freeman, H. (1961). On the encoding of arbitrary configurations. *IEEE Transaction on Electronic Computers*, EC-10:260–268.

Gilmore, R. (1981). *Catastrophe theory for scientists and engineers*. Dover Publications, Inc.

Gonzales, R. C. and Woods, R. E. (1993). *Digital Image Processing*. Addison-Wesley Publishing Company, 3rd. edition.

Granlund, G. H. (1972). Fourier preprocessing for hand print character recognition. *IEEE Transactions on Computers*, 21(2):195–201.

Griffin, L. D. and Colchester, A. C. F. (1995). Superficial and deep structure in linear diffusion scale space: Isophotes, critical points and separatrices. *Image and Vision Computing*, 13(7):543–557.

Grill, S., Zykov, V. S., and Müller, S. C. (1996). Spiral wave dynamics under pulsatory modulation of excitability. *J. Phys. Chem.*, 100:19082–19088.

Haar Romeny, B. M. t., editor (1994). *Geometry-Driven Diffusion in Computer Vision*. Number 1 in the series Computational Imaging and Vision. Kluwer Academic Publishes, Dordrecht, Netherlands.

Haar Romeny, B. M. t., Niessen, W. J., Wilting, J., and Florack, L. M. J. (1994). Differential structure of images: Accuracy of representation. In *Proc. First IEEE Internat. Conf. on Image Processing*, pages 21–25, Austin, TX. IEEE.

Hadamard, J. (1902). Sur les problèmes aux Dérivées partielles et leur signification physique. *Bulletin of University of Princeton*, 13:49–62.

Halsey, T. C., Jensen, M. H., Kadanoff, L. P., Procaccia, I., and Shraiman, B. I. (1986). Fractal measures and their singularities:

The characterization of strange sets. *Physical Review A*, 33:1141–1151.

Hanusse, P., Bastardie, E., and Vidal, C. (1990). Using pattern recognition techniques in the quantitative analysis of the target patterns of the BZ reaction. In Gray, P., Nicolis, G., Baras, F., Borkmans, P., and Scott, S. K., editors, *Spatial Inhomogeneities and Transient Behaviour in Chemical Kinetics*, Proceedings in nonlinear science, pages 371–382. Manchester University Press, Manchester.

Hentschel, H. G. E. and Procaccia, I. (1983). The infinite number of generated dimensions of fractals and strange attractors. *Physical 8D*, 8:435–444.

Horn, B. K. P. and Weldon Jr., E. J. (1986). Filtering closed curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(5):665–668.

Hornik, K. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366.

Hummel, R. and Moniot, R. (1989). Reconstructions from zero-crossings in scale-space. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(2):2111–2130.

Iijima, T. (1962). Basic theory on normalization of a pattern (in case of typical one-dimensional pattern). *Bulletin of Electrotechnical Laboratory*, 26:368–388. (in Japanese).

Iijima, T. (1971). Basic equation of figure and observational transformation. *Transactions of the Institute of Electronics and Communication Engineers of Japan*, 54-C(7):37–38. English Abstracts.

Iijima, T. (1972). A theoretical study of the pattern identification by matching method. In *First USA-Japan Computer Conference*, pages 42–48.

Illner, R. and Neunzert, H. (1993). Relative entropy maximization and directed diffusion equations. *Math. Meth. Appl. Sci.*, 16:545–554.

Jägersand, M. (1995). Saliency maps and attention selection in scale and spatial coordinates: An information theoretic approach. In *Fifth International Conference on Computer Vision*, pages 195–202. IEEE Computer Society Press.

Jaynes, E. T. (1968). Prior probabilities. *IEEE Transaction on Systems Science and Cybernetics*, ssc-4(3):227–241.

Jensen, F. G., Sporring, J., Nielsen, M., and Sørensen, P. G. (1998). Analysing the dynamics of target and spiral waves by image processing techniques. Technical Report DIKU-98/16, Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2200 Copenhagen East, Denmark.

Johansen, P. (1997). Local analysis of image scale space. In (Sporring et al., 1997), chapter 10, pages 139–148.

Kitchen, L. and Rosenfeld, A. (1982). Gray-level corner detection. *Pattern Recognition Letters*, 1:95–102.

Koenderink, J. J. (1984). The structure of images. *Biological Cybernetics*, 50:363–370.

Koenderink, J. J. (1990). *Solid Shape*. M. I. T. Press, Cambridge.

Kolmogorv, A. N. (1965). Three approaches to quantitative definition of information. *Problemy Paredachi Informatsii*, 1:3–11.

Krim, H. and Brooks, D. H. (1996). Feature-based segmentation of ECG signals. In *IEEE International Symposion on Time Frequency/Scale Analysis*, Paris, France.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *Ann. Math. Stat.*, 22:79–86.

Lechleiter, J. and Clapham, D. (1992). Molecular Mechanisms of Intracellular Calcium Excitability in X. laevis Oocytes. *Cell*, pages 284–294.

Lindeberg, T. (1994). *Scale-Space Theory in Computer Vision*. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Boston, USA.

Lindeberg, T. and Li, M.-X. (1997). Segmentation and classification of edges using minimum description length approximation and complementary junction cues. *Computer Vision and Image Understanding*, 67(1):88–98.

Lowe, D. G. (1988). Organization of smooth image curves at multiple scales. In *Proc. 2nd ICCV*, pages 558–567.

Mokhtarian, F. and Mackworth, A. K. (1992). A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):789–805.

Nielsen, M. (1995). *From Paradigm to Algorithms in Computer Vision*. PhD thesis, DIKU, Datalogisk Institut ved Københavns Universitet, Copenhagen, Denmark. DIKU-95-8.

Niessen, W. J., Vincken, K. L., Weickert, J., and Viergever, M. A. (1997). Nonlinear multiscale representations for image segmentation. *Computer Vision and Image Understanding*, 66:233–245.

Niessen, W. J., Vincken, K. L., Weickert, J., and Viergever, M. A. (1998). Three-dimensional MR brain segmentation. In *Proc. Sixth Int. Conf. on Computer Vision (ICCV '98, Bombay, Jan. 4-7, 1998)*, pages 53–58.

Nohre, R. (1994). *Some Topics in Descriptive Complexity*. PhD thesis, Linköbing University.

Oliensis, J. (1993). Local reproducible smoothing without shrinkage. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3):307–312.

Olsen, O. F. (1996). Multi-scale segmentation of grey-scale images. Technical report, Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2200 Copenhagen East, Denmark. DIKU-rapport 96/30.

Olsen, O. F. (1997). Multi-scale watershed segmentation. In (Sporring et al., 1997), pages 191–200.

Oomes, A. and Snoeren, P. R. (1996). Structural information in scale-space. In Johansen, P., editor, *Proceedings of the Copenhagen Workshop on Gaussian Scale-Space Theory*, pages 48–57, Universitetsparken 1, DK-2100 Copenhagen, Denmark. DIKU-96/19.

Osher, S. and Sethian, S. (1988). Fronts propagating with curvature dependent speed: algorithms based on the Hamilton-Jacobi formalism. *J. Computational Physics*, 79:12–49.

Peleg, S., Naor, J., Hartley, R., and Avnir, D. (1984). Multiple resolution texture analysis and classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:518–523.

Peng, B., Scott, S. K., and Showalter, K. (1994). Period doubling and chaos in a three-variable autocatalator. *J. Phys. Chem.*, 94(1):5243.

Perona, P. and Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639.

Picard, R., Graczyk, C., Mann, S., Wachman, J., Picard, L., and Campbell, L. (1995). Vistex. via ftp:whitechapel.media.mit.edu. Copyright 1995 Massachusetts Institute of Technology.

Ramer, U. (1972). An iterative procedure for the polygonal approximation. *Computer Graphics and Image Processing*, 1(3):244–256.

Rasmussen, C. E. (1993). Generalization in neural networks. Master's thesis, Technical University of Denmark.

Reddy, M. R., Nagy-Ungvarai, Z., and Müller, S. (1994). Effect of Visible light on Wave Propagation in the Ruthenium-Catalyzed

Belousov-Zhabotinsky Reaction. *The Journal of Physical Chemistry*, 98:12225–12259.

Rényi, A. (1976a). On measures of entropy and information. In (Turán, 1976), pages 565–579. (Originally: Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, 1961, pp. 547–561, University of California Press).

Rényi, A. (1976b). On the foundations of information theory. In (Turán, 1976), pages 304–318. (Originally: Rev. Inst. Internat. Stat., 33, 1965, pp. 1–14).

Rényi, A. (1976c). Some fundamental questions of information theory. In (Turán, 1976), pages 526–552. (Originally: MTA III. Oszt. Közl., 10, 1960, pp. 251–282).

Rieger, J. (1992). Generic properties of edges and "corners" on smooth greyvalue surfaces. *Biological Cybernetics*, 66:497–502.

Rieger, J. (1995). Generic evolution of edges on diffused greyvalue surfaces. *Journal of Mathematical Imaging and Vision*, 5:207–217.

Rissanen, J. (1983). A universal data compression system. *IEEE Transactions on Information Theory*, IT-29:656–664.

Rissanen, J. (1989). *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore.

Rissanen, J. J. (1996). Fisher information and stochastic complexity. *IEEE Transactions on Information Theory*, 42(1):40–47.

Rodriguez, J. and Vidal, C. (1989). Measurement of Convection Velocities in "Mosaic" Patterns. *J. Phys. Chem.*, 93(7):2737–2740.

Rohr, K. (1992). Modelling and identification of characteristic intensity variations. *Image and Vision Computing*, 10(2):66–76.

Rohr, K. (1994). Localization properties of direct corner detectors. *Journal of Mathematical Imaging and Vision*, 4:139–150.

Rosin, P. L. and Venkatesh, S. (1993). Extracting natural scales using fourier descriptors. *Pattern Recognition*, 26(9):1383–1393.

Rosin, P. L. and West, G. A. W. (1995). Nonparametric segmentation of curves into various representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12):1140–1153.

Sahoo, P., Wilkins, C., and Yeager, J. (1997). Threshold selection using Renyi's entropy. *Pattern Recognition*, 30:71–84.

Sapiro, G. and Tannenbaum, A. (1995). Area and length preserving geometric invariant scale-spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):67–72.

Sethian, J. A. (1996). *Level Set Methods*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press.

Shannon, C. E. and Weaver, W. (1949). *The Mathematical Theory of Communication*. The University of Illinois Press, Urbana.

Siegert, F. and Weijer, C. (1992). Analysis of optical density wave in propagation and cell movement in the cellular mould Dictyostelium discoideum. In Swinney, H. and Krinsky, V., editors, *Waves and Patterns in Chemical and Biological Media*, volume 49 of *Physica D*, pages 224–232. First MIT Press, Elsevier Science Publishers,B.V.Amsterdam,the Netherlands.

Simonoff, J. S. (1996). *Smoothing Methods in Statistics*. Springer Series in Statistics. Springer-Verlag.

Solomonoff, R. J. (1964). A formal theory of inductive inference. part I and I. *Information and Control*, 7:1–22,224–254.

Sporring, J. (1995). Pruning with minimum description length. In *Proceedings of the 5. Scandinavian Conference on Artificial Intelligence (SCAI'95)*, pages 157–168, Trondheim, Norway.

Sporring, J. (1996). The entropy of scale-space. In *Proceedings of 13'th International Conference on Pattern Recognition (ICPR'96)*, volume I, pages 900–904, Vienna, Austria.

Sporring, J. (1997). A prior of saliency based pruning algorithms. Technical Report DIKU-97/8, Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2200 Copenhagen East, Denmark.

Sporring, J. (1998). A piecewise polynomial blob representation. Technical report, IBM, Almaden Research Center, San Jose, California, USA. (Confidential).

Sporring, J., Nielsen, M., Florack, L., and Johansen, P., editors (1997). *Gaussian Scale-Space Theory*. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Sporring, J., Nielsen, M., Weickert, J., and Olsen, O. F. (1998). A note on differential corner measures. In *Proceedings of 14'th International Conference on Pattern Recognition (ICPR'96)*, Brisbane, Australia.

Sporring, J. and Weickert, J. (1997). On generalized entropies and scale-space. In *Scale-Space Theory in Computer Vision, Proc. 1st International Conference*, volume 1252 of *Lecture Notes in Computer Science*, pages 53–64, Utrecht, The Netherlands.

Svarer, C., Hansen, L. K., and Larsen, J. (1993). On design and evaluation of tapped-delay neural network architectures. In Berenji, H. R., editor, *Proceedings of the 1993 IEEE Int. Conference on Neural Networks (ICNN93)*, pages 45–51.

Turán, P., editor (1976). *Selected Papers of Alfréd Rényi*. Akadémiai Kiadó, Budapest.

Véhel, J. L. (1998). Introduction to the multifractal analysis of images. In Fisher, Y., editor, *Fractal Image Encoding and Analysis*, NATO ASI, chapter 17. Springer-Verlag.

Véhel, J. L. and Vojak, R. (1998). Multifractal analysis of choquet capacities: Preliminary results. *Advances in Applied Mathematics*, 20(1):1–43.

Wallace, C. S. and Boulton, D. M. (1968). An information measure for classification. *Computer Journal*, 11(2):185–194.

Weickert, J. (1998). *Anisotropic diffusion in image processing*. Teubner Verlag, Stuttgart.

Weickert, J., Haar Romeny, B. M. t., and Viergever, M. A. (1998). Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3):398–410.

Weickert, J., Ishikawa, S., and Imiya, A. (1997a). On the history of Gaussian scale-space axiomatics. In (Sporring et al., 1997), chapter 4, pages 45–59.

Weickert, J., Zuiderveld, K. J., ter Haar Romeny, B. M., and Niessen, W. J. (1997b). Parallel implementations of AOS schemes: A fast way of nonlinear diffusion filtering. In *Proc. 1997 IEEE International Conference on Image Processing (ICIP-97, Santa Barbara, Oct. 26-29, 1997)*, volume 3, pages 396–399.

Wiener, N. (1948). *Cybernetics*. Wiley, New York.

Winfree, A., Caudle, S., Chen, G., McGuire, P., and Szilagyi, Z. (1996). Quantitative optical tomography of chemical waves and their organizing centers. *Chaos*, 6(4).

Witkin, A. P. (1983). Scale–space filtering. In *Proc. 8th Int. Joint Conf. on Artificial Intelligence (IJCAI '83)*, volume 2, pages 1019–1022, Karlsruhe, Germany.

Yuille, A. L. and Poggio, T. A. (1983). Fingerprint theorems for zero crossings. Technical Report A.I. Memo 730, M. I. T. Press.

Yuille, A. L. and Poggio, T. A. (1986). Scaling theorems for zero crossings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:15–25.

Zuniga, O. A. and Haralick, R. (1983). Corner detection using the facet model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 30–37, Arlington, VA, USA.

List of Publications

This is the pruned list of my publications. I've only included works in their most complete form.

Books Edited

- J. Sporring, M. Nielsen, L. Florack, and P. Johansen (eds.):
  "Gaussian Scale-Space Theory",
  1997, ISBN 0-7923-4561-4, Computational Imaging and Vision, Kluwer Academic Publishers.

Chapters in Books

- J. Sporring and M. Nielsen:
  "Direct estimation of First Order Optic Flow",
  1995, In G. Borgefors (ed.): Theory And Applications of Image Analysis II - Selected Papers from the 9th Scandinavian Conference on Image Analysis, pp. 225-238, World Scientific.

Papers in Journals

- J. Sporring, J. Weickert:
  "Information Measures in Scale-Spaces",
  IEEE Transaction on Information Theory, April 1999 (to appear).

Papers Submitted to Journals

- J. Sporring, M. Nielsen, J. Weickert, and O. F. Olsen:
  "A Note on Differential Corner Measures",
  January 1998.

- F. G. Jensen, J. Sporring, M. Nielsen, and P. G. Sørensen:
  "Tracking Target and Spiral Waves",
  August 1998.

Papers in Conference Proceedings

- J. Sporring, M. Nielsen, J. Weickert, and O. F. Olsen:
  "A Note on Differential Corner Measures",
  August 1998, In Proceedings of 14'th International Conference on Pattern Recognition (ICPR'96), Brisbane, Australia.

- J. Sporring and J. Weickert:
  "On Generalized Entropies and Scale-Space",
  July 1997, In Bart ter Haar Romeny, Luc Florack, Jan Koenderink, and Max Viergever (Eds.): "Scale-Space Theory in Computer Vision", pp. 53-64, Lecture Notes in Computer Science, LNCS 1252, Springer.

- J. Sporring:
  "The Entropy of Scale-Space",
  August 1996, In Proceedings of 13'th International Conference on Pattern Recognition (ICPR'96), Vienna, Austria.

- J. Sporring and M. Nielsen:
  "Direct estimation of Time To Contact",
  June 1995, In Proceedings of the 9th Scandinavian Conference on Image Analysis, pp. 941-948, Uppsala, Sweden. A

- J. Sporring:
  "Pruning with Minimum Description Length",
  May 1995, In Aamodt and Komorowski (eds.) Proceedings of the 5. Scandinavian Conference on Artificial Intelligence (SCAI'95), pp. 157-168, Trondheim, Norway.

- J. Sporring, A. Møller, and P. Hjæresen:
  "Automatic Recognition of Musical Instruments",
  June 1994, In H. S. Olsen (ed.): Proceedings for Nordic Acoustical Meeting (NAM'94), pp. 237-242, Danish Technological Institute.

Technical Reports and Theses

- J. Sporring:
  "A Piecewise Polynomials Blob Representation",
  August 1998, Tech. Report, Almaden Research Center, IBM, California, USA.

- J. Sporring:
  "A Prior of Saliency Based Pruning Algorithms",
  June 1997, Tech. Report DIKU-97/8, Department of Computer Science, University of Copenhagen, Denmark.

- J. Sporring:
  "Statistical Aspects of Generalization in Neural Networks",
  January 1995, Master Thesis, Department of Computer Science, University of Copenhagen, Denmark.

Sammenfatning (Danish)

At måle og modellere billeder har været det centrale tema i denne afhandling. Vi har taget udgangspunkt i en målingsteori kendt som det Lineære Skalarum, og argumenteret for denne metodes anvendelighed i billedbehandling indenfor konventionelle billedhandlingsproblemer. Med den som udgangspunkt har vi også belyst fundamentale problemstillinger indenfor afhandlingens andet hovedtema: Informationsteori. Informationsteori blev først benyttet som et modelleringsværktøj, og derefter analyseret for dens skaleringsegenskaber ikke blot i det lineære skalarum, men også i en række ikke-lineære skalarum.

Et billede består af pixels arrangeret i et kvadrat. Pixlerne er hver især resultat af en måling, f.eks. aktiveres hvert billedelement i almindelige lommekameraer af det indfaldne lys, så længe lukkemekanismen er åben. Billedbegrebet er meget bredt, således kan en-, tre- og højeredimensionelle data samt statistiske sandsynlighedsfordelinger betragtes som billeder.

Billedbehandling er altså metoder der anvender og behandler billeder. Nogle gange er det en fordel at betragte et billedbehandlingsprogram som et, der tager et billede som indata og som udata producerer et billede af samme format. Dog vil sådan et billedbehandlingsparadigme kun meget klodset kunne håndtere spørgsmål som: Er det et billede af et hus? Hertil ville man forvente et ja/nej svar, og ikke et billede. Skalarum falder derimod klart indenfor billedbehandlingsparadigmet: Et skalarum er en udvidelse af billedet med en skalaparameter, således at skridt langs skalaparameteren ændrer billedet i alle pixlerne efter den lokale struktur. Altså billede ind og billede ud. De skalarum, der blev er behandlet i denne afhandling, kan alle skrives som en diffusionsligning. Specielt det lineære skalarum har været i fokus. Det lineære skalarum har den egenskab, at det kan omskrives til en foldning af det oprindelige billede med en Gauss funktion (også kendt under navnet Normal fordeling), hvor variansen er proportional med skalaparameteren. Dette er en nyttig egenskab til billedanalyse opgaver.

Objekter i billeder har ikke på forhånd nogen fastsat størrelse, og det er her at skalarum har sin berettigelse. Istedet for at skrive en algoritme til behandling af objekter for hver mulig størrelse, kan man med hjælp af skalarum nøjes med at skrive en enkelt algoritme til behandling af de mindste objekter. Derefter vil skalaparameteren automatisk og på en entydig måde modificere billedet, så store objekter bliver små, og små objekter forsvinder. F.eks. definerede vi i kapitel 3 centrum for spiraler ud fra deres lokale og globale struktur. Da vi ikke på forhånd var klar over, præcist hvilke størrelser programmet skulle tage højde for, indlejrede vi programmet i det lineære skalarum. Til sidst brugte vi de to skalaparametre, en for den lokale og en for den globale struktur, til at finjustere programmet til de faktiske billedata. Det omvendte var tilfældet i kapitel 5, hvor vi benyttede den fulde skalastruktur til at analysere billeder med. Målet var at komprimere billeder af sort/hvide bogstaver (klatter) ud fra deres geometriske struktur. Strukturen for sort/hvide klatter er særlig enkel, idet man kun behøver at beskæftige sig med overgangene mellem sort og hvid. Ydermere vil alle disse overgange kunne grupperes i lukkede kurver kaldet konturer. Trods alt dette indbefatter komprimeringen af klatter et større søgningsarbejde, da antallet af mulige konturbeskrivelser er meget stort. Til komprimering ønsker man netop en af de korte beskrivelse. Skalarum var for komprimering af klatterne en uvurderlig hjælp til at reducere søgetiderne med. Vha. skalarum blev der skrevet et billedbehandlingsprogram, som analyserede konturerne ud fra deres skalastruktur. Derefter skulle kun de mest sandsynlige beskrivelser gennemsøges istedet for alle mulige.

Analyse af skalastrukturer i billeder har vidtrækkende muligheder og anvendelser. For eksempel er der en klasse af billeder kaldet teksturbilleder, som udmærker sig ved at være regulære i deres rumlige struktur, samt tilnærmelsesvis uafhængige af den perspektiviske projektion. Et teksturbillede kunne f.eks. være et nærbillede af en dørmåtte. Det første trin i en analyse af teksturbilleder vil som oftest være en estimering af størrelsesforholdene, dvs. skalastrukturen. Her er altså tale om et eller flere tal, der indikerer hvor store eller små strukturer, der er tilstede i billedet, svarende til om man er langt fra eller tæt på dørmåtten. Der er selvsagt mange funktioner, som tager et billede som indata og giver et tal som udata. Vi har valgt at analysere generaliserede entropier i det ovennævnte perspektiv. Der er to grunde hertil: For det første er de kompleksitetsmål i informationsteoretisk forstand, og for det andet indeholder de ingen information om de rummelige forhold imellem pixlerne. Til gengæld forholder de generaliserede entropier sig til billedet på samme måde som histogrammet af pixel værdier, gråtonehistogrammet. Studiet af skalastrukturen af de generaliserede entropier er altså også studiet af skalastrukturen af gråtonehistogrammet. Det skal understreges, at det ikke er entropierne af gråtonehistogrammet, som analyseres, men entropierne af billedet selv. I kapitel 7 gennemgik vi den matematiske struktur af de generaliserede entropier ikke blot for det lineær skalarum men også for de ikke lineære skalarum. Strukturen viste sig at være særlig simpel, og analysen er øjensynlig anvendelig for flere forskellige billedbehandligsproblemer relateret til skala.

Studiet af de generaliserede entropier har rejst et væsentligt spørgsmål: Hvilke billeder vil have identisk spektrum af entropier paa alle skalaer? Begrænser man sig til det oprindelige billede er svaret enkelt, idet spektret af generaliserede entropier er ækvivalent med gråtonehistogrammet. Altså vil alle billeder med samme gråtonehistogram have samme spektrum. Derimod er det endnu ikke lykkedes os at finde et definitivt svar for alle skalaer. Som første skridt analyserede vi i kapitel 8 gråtonehistogrammet af endimensionelle funktioner i grænsen af meget fin opløsning, dvs. det kontinuerte gråtonehistogram. For diskrete gråtonehistogrammer kan en hvilken som helst funktionsværdi ombyttes med en anden, uden at gråtonehistogrammet ændres. I det

kontinuerte gråtonehistogram er der derimod en direkte sammenhæng mellem gråtonehistogrammet og funktionens differentialkvotient. Det er dermed ikke muligt at ændre funktionen væsentligt uden også at ændre gråtonehistogrammet. Især for de analytiske funktioner, der i en lille omegn er injektive, blev det bevist at kun de endimensionelle funktioner, som er translationer og/eller spejlinger af hinanden, har identiske kontinuerte gråtonehistogrammer. For de funktioner, som ikke har injektive omegne, fandt vi ikke et tilsvarende bevis. Det er dog sikkert, at mulige variationer af disse funktioner er stærkt begrænset af det kontinuerte gråtonehistogram.

Afhandlingens sidste videnskabelige arbejde, kapitel 9, beskæftigede sig ligesom det forrige udelukkende med informationsteori. Visse modeller for data har et overmåde stort antal parametre. Selvom disse modeller kan indstilles til at stemme godt overens med et givent datasæt, er det langt fra sikkert, at de dermed har fanget trenden i data – at de generaliserer godt. Som i kapitel 5 kan man benytte informationsteori til at vælge modeller for data ved at afveje kompleksiteten af modellen kontra dens afvigelse på data. Ofte viser det sig, at færre parametre i en model er bedre end mange, idet for mange frihedsgrader tenderer til at blive brugt til at modellere støj. I kapitel 9 studerede vi en metode kaldet Optimal Brain Damage, som har en meget generel tilgangsvinkel til modelvalg ud fra dataafvigelsesfunktionen. Metoden benytter sig ikke eksplicit af en afvejning mellem modellens kompleksitet og dens afvigelse på data, men tilsyneladende kun af afvigelsen. Det viste sig dog, at metoden har en indbygget symmetri, således at der findes en ikke triviel afvejning af model og afvigelse, hvor kun afvigelsen får betydning for modelvalget. Dermed kan man sige, at metoden benytter en ikke triviel og implicit afvejning af model- og datakompleksitet.

Afhandlingen blev indledt med en anvendelse som illustrerede nyttigheden af at opdele problemanalysen i en måle- og en modelfase. Som måleteori har vi benyttet skalarum, og til modelvalg informationsteori. Som basis for modelvalg har vi således brugt informationsmål. Ergo er modelvalg også en til tider meget simpel måling. Når man analyserer begreberne mere nøje, indser man, at en opdeling i måling og modellering giver bedst mening ved praktiske problemanalyser. Omvendt

kan en sådan teoretisk analyse give inspiration til nye mål at basere modelvalg på som illustreret ved de kontinuerte histogrammer.

Index