Orientation Analysis; multi-valuedness and shape descriptors

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op 17 mei 2010 om 10.00 uur

door

Frank Gerardus Adrianus FAAS

doctorandus in de sterrenkunde
geboren te Breda.

Dit proefschrift is goedgekeurd door de promotor:
Prof. dr. ir. L.J. van Vliet

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. L.J. van Vliet,	Delft University of Technology, promotor
Prof. dr. E. Bengtsson,	Uppsala University, Sweden
Prof. dr. H.J. Tanke,	Leiden University Medical Center
Prof. dr. ir. W.J. Niessen,	Erasmus MC
Prof. dr. ir. J. Biemond,	Delft University of Technology
Prof. dr. ir. P.P. Jonker,	Delft University of Technology
Dr. Th. Gevers,	University of Amsterdam

This work was partly supported by the Rolling Grants program of the Foundation for Fundamental Research on Matter, project number 94RG12.

Advanced School for Computing and Imaging

This work was carried out in graduate school ASCI.
ASCI dissertation series number 199.

ISBN: 978-90-9025319-0

Contents

Introduction

1

In the past decades automated imaging systems have become common tools in a wide range of settings, i.e. not only in science and industry but also in our daily life. With increasing amounts of generated data and utilisation of imaging systems the need for automated image processing and analysis techniques grows day by day. For an example of these systems one can think of e.g. photo and video cameras, CT and MRI scanners, microscopes and telescopes. All these systems require somekind of interpretation by human or machine.

A crude division of the field of image processing is to divide it into a qualitative and quantitative part. On the qualitative side the human visual system (HVS) excels and is still unrivalled after half a century of scientific progress. Image processing can assist the HVS to make interpretation of images easier by transforming the data in such a way that it better matches the characteristics of the HVS. In other words, image processing should enhance desired features and suppress the others, especially noise. Examples of these techniques are non-linear contrast enhancements, multi-frame super-resolution reconstruction techniques and (non)-linear filter schemes. When, however, quantitative aspects become more important, computerised methods are superior to the HVS. Such superiority arises in the first place from the fact the HVS is not designed to measure but to recognise. Therefore applications that require measurements with high accuracy, repeatability or objectivity benefit the most from automated image processing. Applications that provide data in need for thousands of measurements just benefit from the savings in man-power and spares the operator the boredom of repeated tasks.

1.1 Problem

We recognise that virtual all image information is embedded in intensity and/or spectral transitions giving rise to boundaries. This implicates that regions with little variation in measurement values are essentially void of information. Structural image variations can in overwhelming majority be locally described by lines, edges or surfaces boundaries. In addition to contrast and scale, the orientation of these transitions plays the most important role in many vision, image analysis and recognition tasks. The HVS incorporates brain cells that give selective responses to each of these transitions individually [14]. Images are thus transformed into a redundant representation in the brain which allows for easier analysis. In the case of highly structured image content or when multiple simple structures are present this enables recognition. In generic image processing only simple neighbourhoods can be handled. A generic filter is not aware that the neighborhood it is applied to may be composed of multiple instances of simple structures. These filters yield a single output value that cannot be interpreted in terms of the constituent simple structures. Note that local averaging filters give meaningful results in homogeneous regions that are hampered by noise, but fail when the filter support includes a transition, i.e. edge or line. A first derivative filter is the proper tool to process/analyze edges, but fail when multiple edges (including lines) are within the filter kernel. To be able to apply these generic filters to "complex" or multimodal neighbourhoods, i.e. regions with multiple oriented simple structures such as crossing lines, the unentanglement into simpler neighbourhoods is an attractive option.

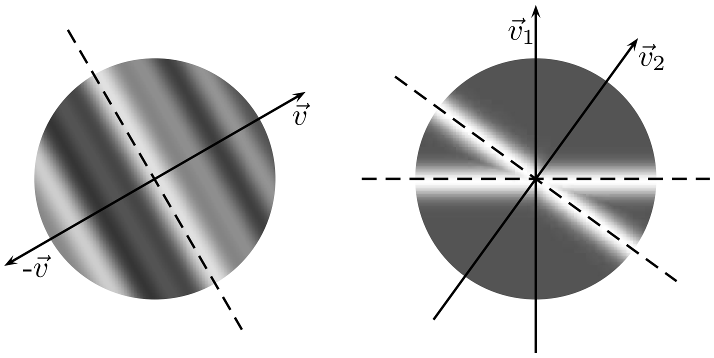
In this thesis we will first focus on quantitative orientation measurements in multimodal regions. This to enable direct recognition or segmentation as well as the application of image processing on these simpler constituencies. Several different new approaches are introduced and compared to existing work. We want to make clear that we do not enhance the image content but try to perform *measurements* such that direct *detection* is possible also in multimodal image regions. Once the underlying simpler image building blocks, mostly lines and edges, are found subsequent image analysis can perform measurements a lot easier. For the application of image processing and analysis we focus on one class of these simpler neighbourhoods: curvilinear objects. To be more specific we aim to automatically analyse DNA molecules imaged in transmission electron microscopy (TEM) and atomic force microscopy (AFM). As such the two main technical topics in this thesis are:

- multi-valued orientation measurements in complex neighbourhoods, and
- distance transforms and minimum cost path methods for the extraction of the medial axis of curvilinear structures.

Single Orientation Neighbourhoods

Let us first discuss the nature of simple building blocks before we move to complexer neighbourhoods and explain our approaches. In essence, orientation is the axis along which the intensity variations in a local neighbourhood in an image occur. As orientation in practice is a property of a local neighbourhood or window and not a global property, Granlund and Knutsson introduced the concept of simple neighbourhoods [56, 62, 38, 51]. Simple functions or neighbourhoods can be seen as structures which vary along one or more dimensions but are constant in the other dimensions. Examples of such structures are lines, planes and edges. Mathematically a simple neighbourhood with rank r is present in a scalar image $f : \mathbb{R}^n \to \mathbb{R}$ if we can write:

$$f(\boldsymbol{x}) = g(\boldsymbol{v}_1^T \boldsymbol{x}, ..., \boldsymbol{v}_r^T \boldsymbol{x}) \tag{1.1}$$



(a) (b)

Figure 1.1: (a) A simple function with an arbitrary signal along the translation variant axis $\boldsymbol{v}$ and a constant signal along the translation invariant axis (dashed line). (b) Similar as in (a) but with two superimposed simple structures, i.e. two crossing lines.

with g a function of $r < n$ arguments, $\boldsymbol{x} \in \mathbb{R}^n$ the Euclidean coordinate and $\boldsymbol{v}_i$ the column vectors representing the r non-constant dimensions. As illustrated by Fig. 1.1(a), the direction of the vector representing the non-constant dimensions of a simple function is not unique. To avoid this ambiguity the property of orientation is introduced:

> The orientation of a simple n-D neighbourhood can be specified by either the $n - r$ orthogonal axes in which the neighbourhood is translation invariant or by the r axes that are orthogonal to the translation invariant axes.

For example the orientation of a line in 3D is given by its axis of symmetry but also by a set of axes spanning the plane perpendicular to this axis. The reverse holds for planes, e.g. the axes spanning the plane represent the translation invariant axes but more intuitive is the use of its complement i.e. the normal to the plane. As in this thesis the highest dimensionality we consider is 3D, we always choose a single axis representation. In higher dimensions, objects exist with a rank higher than one and a complement which also has a rank higher than one, which means that multiple axes are needed for the description of the orientation of such a structure (see Fig. 1.2).

Orientation estimation in simple neighbourhoods has been considered in the literature for a long time, see e.g. [9, 49, 40, 46, 105]. The measurement of orientation can be the final goal but it can also serve as a starting point for further analysis steps like curvature estimation [53, 108, 111], directed diffusion [154, 104, 45] or texture characterisation [32]. Historically, image processing algorithms are designed for simple neighbourhoods. Therefore numerous methods show an excellent performance on isolated structures but few of them can cope with more complex neighbourhoods where the assumption of isolation no longer holds, i.e. in regions where multiple image primitives overlap, touch or cross.

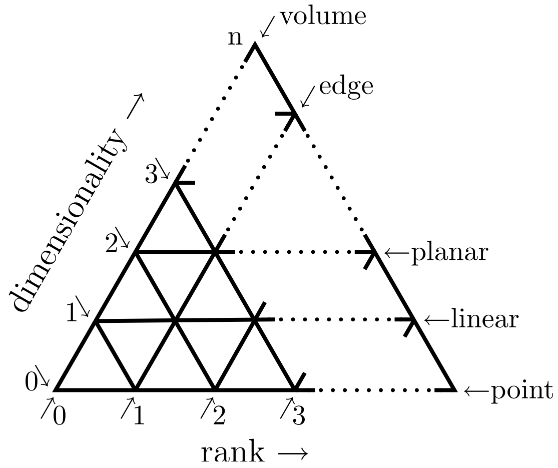


Figure 1.2: Diagram relating simple functions to structures and dimensionality.

1.2 Approaches

1.2.1 Multi orientation estimation

So far we considered the presence of a single structure in a local neighbourhood. Obviously not all regions in an image can be represented by a single simple function. We can, however, describe most regions as a composition of these simple structures each with its own orientation, see Fig. 1.1(b). In practice, the number of orientations that cross in a local neighbourhood is *small*. If considering random placement of line-like structures in 2D for the moment, then it is clear that these lines will intersect, but the chance that more than two cross in the same point is small. As the dimensionality increases the likelihood of crossing events decreases further. Hence, multi-orientation estimation in images is only considered for a few (2,3,4) overlapping image primitives.

Existing literature on multi orientation analysis can roughly be divided in two classes: a tensor based and a filterbank based approach. The first class builds upon the structure tensor introduced for single orientation analysis by Zenzo [161], Bigün and Granlund [8] and Kass and Witkin [67]. Only recently, efforts have been made to extent this approach to multiple orientations. These efforts have followed two paths, either by introducing higher order tensors were the traditional tensor is often a limited case or by a non-linear averaging approach which leads to non linear tensors. Shizawa and Iso [130] introduced the concept of ideally* overlapping image primitives followed by Aach et al. [1] and Mota et al. [92]. From the assumption that the underlying primitives are translation invariant they derived a minimisation criterion. They translated this criterion in a tensor representation, where the eigenvectors are mixed orientation vectors. Although the principle is easily extended to more than two overlapping patterns there is so far no efficient way of decomposing the mixed vectors into the orientation vector of the underlying image primitives. In Herberthson et al. [61] outer products are applied to orientation pairs. The method yields an analytical solution for pairs of orientations. Brox et al. [16] introduced the non-linear structure tensor in which the classical Gaussian regularization is replaced

---

*each primitative is assumed to be truely translation invariant along the orientation axis.

by a discontinuity preserving nonlinear diffusion allowing for orientation estimates in close proximity of e.g. crossings and junctions. Moltz et al. [90] constructed a histogram of orientations in a neighbourhood where the orientations are calculated with a structure tensor on a smaller scale. The second class in literature on multiple orientation is based on a rotated set of orientation selective filters [55, 49, 7, 40, 41]. These filters are widely used for image processing tasks like detection of bifurcations, crossings and corners, optic flow estimation and line segmentation. To obtain a high angular resolution, however, a huge amount of filters is required. To get a better trade-off between orientation selectivity and computational demands Freeman and Adelson [46] introduced the concept of steerable filters, i.e. the response from a filter for a specific orientation can be calculated as a linear combination of a relative small set of basis filters [6, 99, 88, 60, 51, 4]. Walters [153] was the first to see orientation as a independent variable, i.e. to see orientation as a separate dimension in the image space. In the literature this concept is known under different names: orientation space, orientation scores, orientation channels and orientation bundles. Most authors use a filterbank with orientation selective filters to construct the orientation dimension(s) [160, 66, 18, 51, 153, 46, 84, 45, 44]. Vliet and Verbeek [148] proposed a local Hough transform as basis for the construction of an orientation space. The approach by Michelet et al. [89], not fitting either of the mentioned classes , proposes a recursive iterative approach to minimise the local difference between the image and a neighbourhood model of straight lines.

The first part of this thesis is dedicated to the detection of these multimodal regions and subsequent orientation estimation of its constituents. Each image primitive carries its own information. In complex neighbourhoods, however, this information becomes entangled as the contributions of the respective underlying structures are mixed. Mixing is not only caused by overlap, e.g. also during image formation and image analysis mixing can occur. In unimodal regions this will normally only cause some blurring leaving the orientation information untouched, this is unfortunately not the case in multimodal and highly curved regions. As such the true challenge of orientation measurements in multimodal regions becomes to disentangle the contributions of the underlying structures.

Several approaches to solve this problem are pursued in this thesis, i.e. methods based on

- orientation selective filters: filters which only select a small band of orientations are used to build an orientation histogram expressing the evidence for the presence of each orientation.
- symmetry: the symmetry in the orientation tensor map is utilised to detect and estimate the orientation of saddle points and crossings.
- streamlines: streamlines are used to connect points in complex neighbourhoods to locations where the underlying structure is unimodal.
- orderless-clustering orientation analysis: when each image primitive is analysed separately a distinct orientation can be found. As such analysing the orientations in a multimodal regions should lead to multiple orientation clusters.
- spatio-clustering orientation analysis: similar to the previous approach only the spatial connectivity between pixels is taken into account.

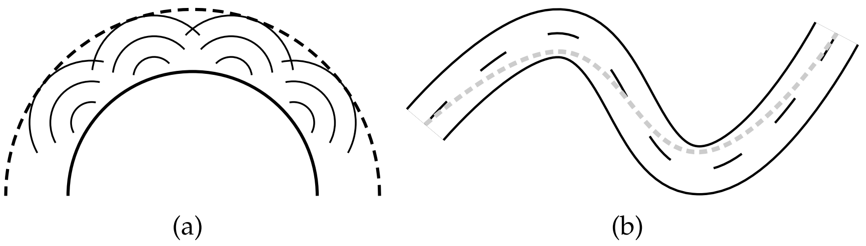


Figure 1.3: (a) Huygens' principle, each source on wave front acts as new source from which the envelope constitutes a new wave front. (b) Fastest path, dotted line, over a winding road with the centreline denoted by the dashed line.

1.2.2 Path extraction from locally one-dimensional objects

Restriction from all possible simple neighbourhoods to one class of these simple neighbourhoods, curved line-like objects, sets the scope for the rest of this work. The goal is to automatically analyse DNA molecules in 2D images acquired by Transmission Electron Microscopy (TEM) and Atomic Force Microscopy (AFM). In the following we also assume that any complex neighbourhoods in the images are already disentangled by suitable methods to resolve these neighbourhoods into simpler substructures. Post-processing of the structures is treated as if they were stored in separated images. This is, however, not necessary if dedicated algorithms are implemented for the path extractions that match the chosen representation.

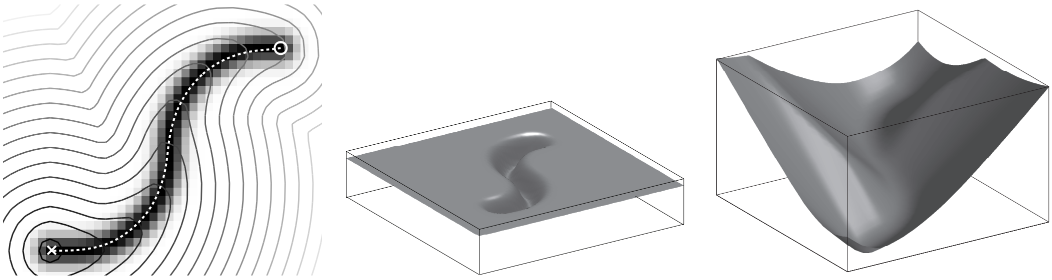
The analysis of DNA properties from 2D images is based on statistical chain descriptors which rely on tracing of the DNA backbone or centreline. Please refer to Fig. 1.5 for a typical TEM image and a path through the centreline of a molecule. To this end the shape of the centreline of the individual molecules is analysed to estimate the statistical properties of the ensemble. The result is used to answer questions with regard to the dimensionality of the adhered DNA in relation to the flexibility of the polymer chain in solution. With this application in mind, we hence forth pose a few requirements for the centreline extraction method. Most importantly, the centreline extraction method should be able to cope with varying contrasts along the structure. The contrast is affected by variations in the density of the staining of the molecule which in turn makes the strand appear as beads on a string. Furthermore, the path extraction should yield a continuous and differentiable representation of the centreline; no model is to be imposed. The method should also not rely on accurate segmentations of the object and finally the method should yield a description of the centreline with a subpixel accuracy and precision. This last requirement is crucial to draw sustained conclusions from properties derived from correlation effects such as the tangent-tangent correlation of the DNA over distances of a few nanometers only, which are imaged onto a few pixels.

The extraction of centrelines of stringlike objects is a commonly encountered problem in image processing. Examples can be found in application areas such as medical image processing in X-ray angiographs, traffic lane surveillance and tracking of tubular structures in biological cells. In the following we review the literature on existing techniques to find the medial axis or centreline of stringlike objects. One of the first methods to be published was the grassfire propagation algorithm by Blum [10] in 1967. Imagine a dry grass patch, the object, which is set on fire on the boundary. The fire is free to propagate inwards with constant speed. Whenever the fire fronts meet, the fire is quenched. The points where the fire was quenched forms the skeleton or medial axis of the object. Also well known is the Voronoi diagram approach. The Voronoi diagram of the object depicts boundaries at equal distance from at least two boundary points, as such the medial axis is a subset of the Voronoi diagram. Probably the most well known method is the thinning approach or binary skeletonisation [125, 143, 12, 131, 64]. This method iteratively peels off a binary shape while preserving its geometrical and topological features. All these methods yield a discrete skeleton of the object either in the form of a set of connected points or as a connected set of line segments. These methods rely on an accurate segmentation of the object prior to processing and are as such generally sensitive to noise.

The following methods do not rely on an accurate segmentation of the object and work directly on the grayvalue representation of the image. This representation is first transformed such that it can function as a potential which is to be minimised subsequently to yield the medial axis. One method is known as the active contour or snake method [68, 22]: starting from some initial contour the method tries to minimise the energy in the contour, where the energy is the sum of an internal and external source. This external energy should be minimal at the actual contour and the internal energy should be minimal for the expected shape of the contour. As such only when the internal energy is constant no model is imposed. Another existing method is the minimum cost path approach [117, 139, 141, 136, 126, 79]. This method minimises a cumulative cost, i.e. the sum of some gray values, along a path between two points, where the cost-function forms a channel in which the local minima lie exactly along the centreline of the object, see Fig. 1.4. The cost function can be just the grayvalue representation of the image or a transformation of it. The latter method is taken as a starting point as it potentially fulfils all our requirements for the centreline extraction for the DNA backbone. The existing methods on minimum cost paths will now be discussed, their shortcomings exposed and we will improve on them to suit our application.

Minimum cost paths

The minimum cost path problem is encountered in various research areas of science and engineering like acoustics, optics, optimisation theory and path planning. Basically it is applicable to any problem which can be formulated as the search for the fastest path through a medium (with a spatial varying speed function). In optics the problem corresponds to finding the path that a ray of light follows between the light source and observer while traversing a medium with a space-variant refractive index. As such Huygens' principle is applicable, i.e. each point on the wavefront can be seen as a secondary source from which the envelope constitutes a new wavefront, see Fig. 1.3(a). The construction of the wavefront for discrete maps, such as a pixel grid, was first solved for the image processing domain by Dijkstra [30]. His idea is applicable to a wider range of problems and allows for an ordered approach. According to the Huygens principle one must use point sources to get the correct solution. Due to difficulties modelling the wavefront correctly as spherical waves and accounting for their interaction this is not yet achieved in image processing. The earliest wavefront models relied on chamfers to describe the local metric which introduces significant deviations from the exact solution [117, 139]. Later, the local



(a) (b) (c)

Figure 1.4: (a). Synthetic curvilinear object, used as cost function to derive the cumulative cost map with respect to **x**. The isochrone lines denote the cumalative cost map (gray value denotes time) and the white dashed line denotes the backpropagated minimum cost path with the start of the descent marked by **o** and the end point by **x**. (b-c). Respectively the cost and cumulative cost map of (a) depicted as height map.

wavefronts were modelled as planar waves [136, 126, 79] which reduced the errors.

In this thesis we introduce a new scheme to model the wave locally as a circular/spherical wave. This yields exact results for space-invariant cost functions where earlier approaches did not, and it yields improved results for space variant cost functions. As stated in the previous section, we are interested in a method which yields an accurate description of the centreline of curvilinear objects. The minimum cost path approach, however, does not find it: they tend to cut the corner. This behaviour is caused by the same reason a race car driver takes the inner bend, i.e. although the maximum attainable speed is lower in the innerbend the path is much shorter resulting in an overall faster arrival at the finish, compare Fig. 1.3(b). Note, however, that on a straight road the car will follow a straight track. Here we present a new iterative approach to solve this problem based on just this observation, i.e. the corner cutting problem is non existent for straight linear structures. In the image domain the cost function is represented by grayvalues and their transitions are locally smooth, disregarding noise and assuming proper sampling of a bandlimited input. The image formation process for TEM and AFM can in our case be modelled as a convolution of the sample with the point spread function or impulse response of the imaging system. This leads inevitably to a blurring/smoothing of the object and the greyvalues at the edges. If the grayvalue landscape is chosen as the cost function of the object it is therefore a smooth valley and not infinitely steep. In turn, this smoothness leads inevitably to the cutting of corners in minimum cost path approaches.

DNA deformations near charged surfaces

Deoxyribonucleic acid (DNA) is a very important cell structural element, which determines the level of gene expression by virtue of its interaction with regulatory proteins. It consists of two polymers each with a sugar phosphate backbone connected into a double helix by the bases, Adenin, Guanin, Cytosin and Thymin. The micromechanics of DNA can be described by several models, the best known of which is the wormlike chain (WLC) model for semi-flexible polymers [76, 77]. It describes the polymer as an elastic homogeneous rod. Note that other models, such

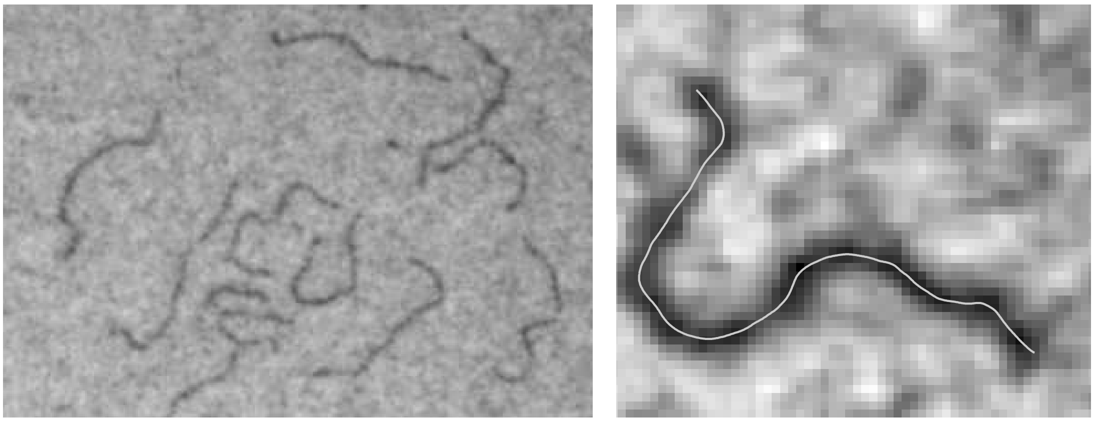


Figure 1.5: (a) DNA molecules stained with uranyl acetate and visualised by transmission electron microscopy. (b) Single molecule with superimposed the extracted centreline of the molecule. The images are kindly provided by Dr. Dmitry Cherny, PhD, Dr.Sc.

as the freely joint chain, which describe the polymer as a chain which only bends at the joints yield the same descriptors given sufficiently long molecules with respect to their stiffness [122, 15]. In the WLC model the polymers are described by their bending rigidity which is the only free parameter of the model. The characteristic bending length scale is called the persistence length P. One can derive various characteristic functions to describe ensembles of these polymers based on $R(l)$ and $\theta(l)$, respectively the point-to-point distance and orientation difference as function of the contour length l. Examples of these functions are the kurtosis, the mean squared end-to-end distance and the orientation correlation function, all of which can be derived by averaging over all possible conformations [76]. All these functions depend on the persistence length P which denotes the stiffness of the polymer. Comparing these descriptors with distributions derived from measurements of DNA molecules allows for the validation or possible rejection of the WLC model in the first place and for measuring the persistence length in the second place. All these statistical descriptors can only be computed *after* the centreline of the DNA backbone has been found from the images. Now it is also evident why it is so important to extract the medial axis to such a high degree of precision and why subpixel localisation is absolutely mandatory.

Here we use TEM and AFM to characterise the flexibility of double-stranded DNA. The analysis of long polymers like DNA with microscopic techniques requires the deposition of the molecules on a surface in a dried state. Adhering the DNA to a surface will cause the loss of one degree of freedom and can possibly induce distortions in the DNA. In the literature there is a fair agreement that the persistence length of DNA in solution is close to 50 nm, given certain solution conditions. There is, however, still discussion on the conformational state of adhered molecules, i.e. if the molecules are in a 2D or 3D state, or even a fractal state. Any deviation of a found persistence length from the literature value is often interpreted as a three-dimensional state of imaged DNA due to surface trapping by lack of a better explanation [63, 138, 93]. The 2D state indicates an equilibrium state close to binding while the 3D state would indicate trapping in which the 3D state before adhering is partly preserved. In this thesis we answer the questions regarding the adhered state of the molecules as well as the question of induced deformations during the adhering process by providing a combination of dedicated image processing and a model describing the statistical behaviour of DNA molecules confined to a plane. By extending the number of statistical quantities, we prove a two-dimensional equilibrium state of immobilised molecules.

1.3 Structure of the thesis

The first part of this thesis focuses on orientation estimation in multi orientation regions. These regions are of utmost importance as single orientation regions cross here and as such carry most of the image information. Traditional image processing produces erroneous output in these regions despite of there importance. Once we can determine where in the image these complex regions are located we can proceed to determine the orientation of the constituents after which a detection step is applied to the (redundant) representation, permitting disentanglement. Although the methods presented in the first part of the thesis vary considerably, all methods adhere to this scheme.

In Chapter 2 we extend the orientation space transform from traditional 2D to 3D. This extends the redundant representation from a 2D+φ to a 3D+(φ, ϑ) representation. Here we use a filterbank based approach and we introduce a set of new orientation selective filters, a multi resolution orientation sampling grid and a novel way to visualise the resulting 5D orientation space. A new crossing detector is introduced in Chapter 3 based on the detection of multiple orientations in a local region. This method utilises symmetry properties in the gradient field of an image to detect and determine the orientations present in crossings and saddlepoints. Chapter 4 introduces yet another method, based on streamlines, to detect junctions or corners and accurately measure the orientations of underlying image primitives. In Chapters 5 and 6 new multimodal orientation analysis methods are presented based on clustering of the gradient fields. The clustering is performed in an orderless fashion (Chapter 5) and in a spatio-orientation approach utilising the orientation pureness of each individual pixel (Chapter 6). These methods disentangle multi-modal regions and assign each pixel a unimodal structure. The unimodal regions allow in turn for standard orientation analysis methods.

A special class of these unimodal regions are curvilinear objects; in our application double stranded DNA (dsDNA). We want to apply image processing to automatically analyse images of DNA obtained by TEM and AFM. The biophysical properties of DNA are classically described by the worm like chain model (WLC) with the persistence length as only free parameter. The formulae describing ensemble averages as the end-to-end distance or the tangent-tangent correlation of this model scale with dimensionality, that is only the persistence length is scaled. We answer the question wether imaging of DNA on a flat support really can give a clear indication of the dimensional state of the deposited DNA. Next to that we investigate the flexibility of DNA over very short distances (< 10 nm) as recent studies indicate a deviation from the WLC here. To this end we developed the necessary tools to find the centreline of curvilinear structures. We start with a review of existing distance transforms in Chapter 7 and then present a distance transform based on a circular wave approximation as an alternative to the well known planar wave approximation which increased the limited accuracy and precision of older methods. To extract centrelines, the ″cutting of corners″ effect associated with minimum cost paths is addressed in Chapter 8. We present an iterative approach that does not suffer from this defect. Finally in Chapter 9 we apply all the above to the flexibility analysis of DNA near charged surfaces.

3D-Orientation space; filters and sampling

2

---

abstract[*]
The orientation space transform is a concept that can deal with multiple oriented structures at a single location. In this paper we extend the orientation space transform to 3D images producing a 5D orientation space (x, y, z, ϕ, θ). We employ a tunable, orientation selective quadrature filter to detect edges and planes and a separate filter for detecting lines. We propose a multi-resolution sampling grid based on the icosahedron. We also propose a method to visualise the resulting 5D space. The method can be used in many applications like (parametric) curve and plane extraction, texture characterisation and curvature estimation.

2.1 Introduction

Three-dimensional images can be seen as compositions of numerous simple structures like planes, textures, edges and lines. Therefore multiple oriented structures can be present at a single point. Here we will describe a method which can deal with such occurrences in 3D. Detectors developed in the past, like the tensor approach [8, 9, 72], can handle single oriented structures but often fail on non-isolated structures. Therefore we present a multi orientation analysis as first proposed by Walters [153] and later implemented in 2D by e.g. Ginkel [51]. In our multi-orientation analysis, we probe to see how much oriented structure is present that exhibits the probe orientation. We filter the image with rotated versions of an orientation-selective template filter m and stack the accumulated evidence in two extra angular dimensions

$$I_m(\boldsymbol{x}, \phi, \theta) = I(\boldsymbol{x}) * m(\boldsymbol{x}; \phi, \theta). \tag{2.1}$$

---

[*] The content of this chapter has been published in: F.G.A. Faas and L.J. van Vliet. 3d-orientation space; filters and sampling. In J. Bigün and T. Gustavsson, editors, *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, LNCS 2749, pages 36–42. Springer, June 29 -July 2 2003 [35].

Here $\boldsymbol{x}$ contains the spatial dimensions, x, y and z. The template orientation is given by ϕ and θ. With ϕ the counterclockwise angle in the xy-plane measured from the positive x-axis, ranging from 0 to 2π, and θ measures the angular distance from the positive z-axis, with θ ranging from 0 to π. The asterisk denotes the convolution operator.

Now let us define orientation for line and plane-like structures. When we draw a line through the centre of a unit sphere we will find two intersection points. These points can be specified by their respective θ and ϕ coordinates. Both pairs of $\{\theta, \phi\}$-coordinates can be used as orientation but to avoid ambiguities we have to drop one pair. So let us now define the orientation of a line as the pair of coordinates for which: $\theta \in [0, \pi/2\rangle$ or $\theta = \pi/2 \wedge \phi \in [0, \pi\rangle$. For a plane we adopt the same formulation in which the line is replaced by the normal to the plane.

Note that this coordinate representation contains discontinuities due to the fact that ϕ and $(\pi/2 - \theta)$ are modulo 2π and $\pi/2$ respectively. However these discontinuities pose no problems for further filtering as they are coordinate discontinuities which can be easily dealt with by applying a standard boundary extension technique. This in contrast to the tensor approach in which the filter output is discontinuous, the output therefore has to be remapped for further processing in most cases [73].

Further we will show a few ways of visualising the results of the orientation space transform.

2.2 3D-Orientation space: filter design

In designing our filters the first thing to realize is the trade-off between orientation selectivity and localisation. By increasing the orientation selectivity the filter becomes more extended and loses localisation, i.e. the filter response changes slowly along the long axes of the filter. We want to treat the orientation and scale selectivity separately, therefore the filter is made polar separable in Fourier space, see e.g. [74],

$$\mathcal{F}\{m\}(\boldsymbol{f}; \phi_i, \theta_i) = M_{rad}(f)\, M_{ang}(\phi, \theta; \phi_i, \theta_i) \qquad \text{with } f = |\boldsymbol{f}|. \tag{2.2}$$

With $\boldsymbol{f} = (f, \phi, \theta)$ the spherical coordinates in Fourier space. Now the radial function specifies the scale and the angular part the selectivity of the filter. Proper sampling and discretization of the resulting orientation space require that the input image, as well as the filters, are band-limited. For correct sampling along the ϕ and θ axes the filters should be radial and angular band-limited [46, 51].

In 2D, lines and edges can be treated equally using quadrature filters. In 3D, there is a similar relation between edges and planes. However the line appears as a new structure which requires separate treatment and has no associated quadrature structure. Therefore we design two filters, one for line like structures and a quadrature filter for planar structures. Furthermore the 3D filters used are generalisations of a 2D filter presented in [51, Chap. 3].

The angular part of the filters is defined as:

$$M_{ang}(\phi, \theta; \phi_i, \theta_i) = M(\boldsymbol{f}, \boldsymbol{\psi_i}) = 2e^{-\frac{1}{2}\frac{\rho^2}{\sigma_s^2}}, \tag{2.3}$$

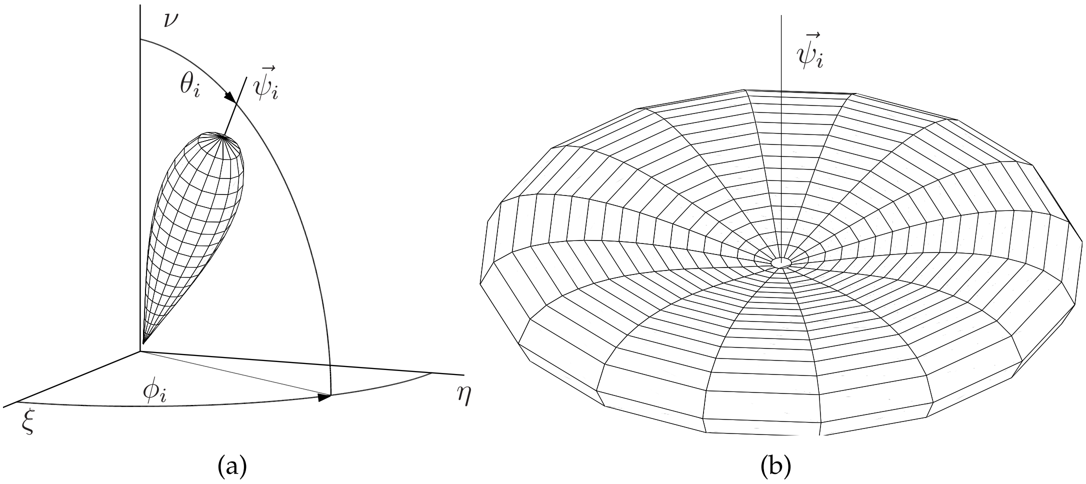


Figure 2.1: (a) Planar quadrature filter M_i , with ξ, η and ν the Cartesian Fourier coordinates and filter orientation angles $\{\phi_i, \theta_i\}$. (b) The same as (a) but for the line filter. The parameters used for these filters: $f_c = 0.2$, $b_f = 0.16$.

where

$$\rho = \begin{cases} \angle(\boldsymbol{f}, \boldsymbol{\psi_i}) & \text{plane filter(spatial)} \\ \pi/2 - \angle(\boldsymbol{f}, \boldsymbol{\psi_i}) & \text{line filter(spatial),} \end{cases} \tag{2.4}$$

with

$$\boldsymbol{\psi_i} = \begin{pmatrix} \cos(\phi_i)\sin(\theta_i) \\ \sin(\phi_i)\sin(\theta_i) \\ \cos(\theta_i) \end{pmatrix}, \tag{2.5}$$

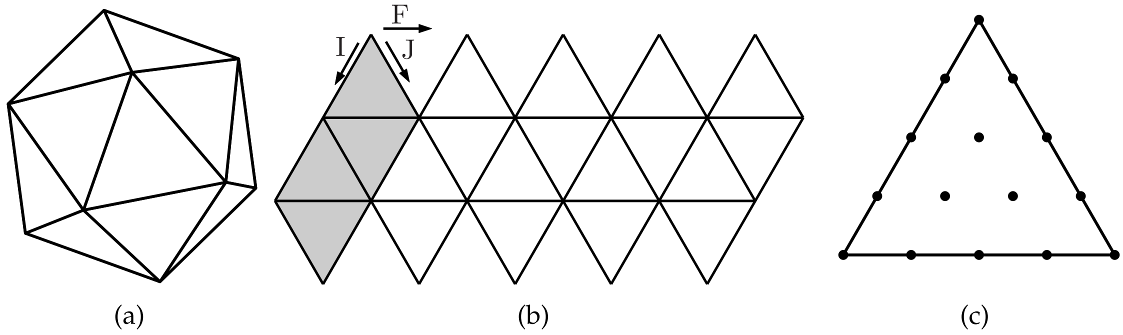
and

$$\sigma = 2\arccos(1 - 2/N). \tag{2.6}$$

The orientation selectivity σ is found by equating the surface area of the unit hemisphere, $S = 2\pi$, with N times the area S_{cone}. Here S_{cone} is given by the intersection of S with a solid cone with opening angle σ. In this formulation the orientation selectivity can be increased by raising N and the sampling distance is approximately 1σ as required by band-limiting the Gaussian [145, Chap. 2]. The quadrature structure for the planar filter only requires that the filter is zero for $\rho > \pi/2$, this is approximately satisfied, as for $N \geq 15$ the 3σ radius lies well within $\pi/2$. Now let us define the radial part of the filter [51],

$$M_{rad} = \left(\frac{f}{f_c}\right)^{\left(\frac{f_c}{b_f}\right)^2} e^{-\left(\frac{f^2 - {f_c}^2}{2{b_f}^2}\right)} \tag{2.7}$$

This Gaussian-like function has a bandwidth b_f and an central frequency f_c. An advantage of this filter over a true Gaussian of bandwidth b_f and centre frequency f_c is a guaranteed zero response to constant signals. The resulting plane filter has a droplet shape and the line filter has a donut shape as footprint in Fourier space, see Fig. 2.1.



(a) (b) (c)

Figure 2.2: (a) Icosahedron, platonic solid with largest number of faces. (b) Unfolded icosahedron. (c) Hexagonal grid at one face of the icosahedron

2.3 3D-Orientation space: sampling grid

Sampling the orientation space by N samples requires filtering image $I(\boldsymbol{x})$ with N rotated copies of an elongated template filter m, such that its principle axis coincides with orientation $\{\phi_i, \theta_i\}$. The result, $I_m(\boldsymbol{x}, \phi_i, \theta_i)$ is a 5D-orientation space. Now let us look at how to distribute the N orientations, $\boldsymbol{\psi_i}$ over one hemisphere, i.e. here the upper half of the unit sphere. We only need to address the upper half as we cannot distinguish between opposite directions.

Ideally the angular distance from each point to its direct neighbours should be constant, at least for 3D volume images like those acquired with MRI. But in cases where the human perception is mimicked this is does not hold, e.g. our perception of speed is not uniform across spatial frequencies [7]. As there is no general solution for the problem of distributing a set of points equidistant over a sphere [119], we have adopted a grid based on the icosahedron inspired by [31]. The icosahedron is the largest of the platonic solids and has 20 identical faces consisting of equilateral triangles and 12 vertices, Fig. 2.2(a) and (b). On each of the faces we impose a hexagonal grid. This grid is then projected on the unit sphere to obtain the orientations (Fig. 2.2). This pixelisation scheme is symmetric in the origin and allows easy indexation (addressing). We can easily change the number of points by imposing a finer/courser hexagonal grid on the faces of the icosahedron. The number of points is given by $N = 5n^2 - 10n + 6$ with n the number of points on a single side of a face ($n \geqq 2$). Furthermore the grid is hexagonal with the exception of the vertex-points of the icosahedron which are pentagonal. Points are indexed with three indices as shown in Fig. 2.2(b). Index F denotes a strip of 4 faces while I and J are subindices on this strip. Note that the points on the border of the strip should be treated differently to avoid multiple indexation of a single orientation. This scheme allows us to easily find the neighboring orientations which is useful for connectivity issues.

2.4 Test experiments and visualisation

Now let us look how we can visualise orientation space. Therefore we investigate a simple image of a fork structure constructed from three line segments with a Gaussian profile of 1σ, see Fig. 2.3(a). The image size is 75^3 voxels and the parameters

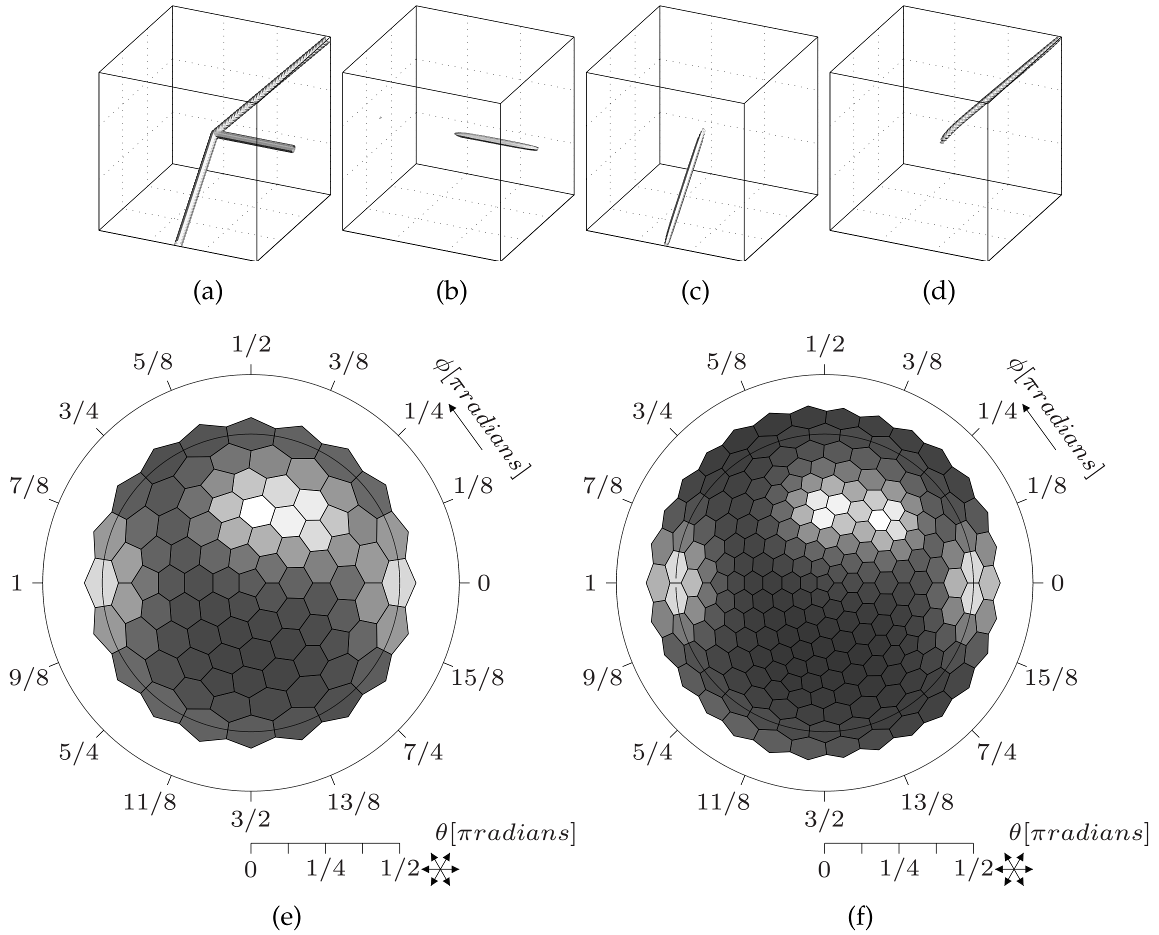


Figure 2.3: (a) isosurface of a fork-structure with a size of 75^3 voxels. (b-d) isosurface of $I_m(\boldsymbol{x}, \phi_c, \theta_c)$ for the three orientations c, with the highest responses.(e-f) Polar Voronoi diagrams of the voxel closest to the intersection point of the three line segments for an orientation space of respectively 81 and 181 sample orientations.

used for calculating the orientation space are (f_c,b_f,N)=(0.2,0.16,181). Figs. 2.3(b-d) show isosurfaces of the response of three filters on the fork image, where the tree filters are those with the highest responses. As can be seen the orientation space response is smooth and contains local orientation information. To use orientation space for segmenting the image, the oriented structures which are in this case line segments, should be resolved. Now we assume two structures can be resolved if the minimum response between the two is less than half the maximum response. This states that the minimum angle between the oriented structures must be larger than $4\sqrt{\ln 2}\sigma$ according to [51]. As the minimum angle of separation in our image is 35° the minimum number of sampling orientations required is 140. Now we will show the output of two angular resolutions with respectively 81 and 181 sampling orientations. Let us inspect the angular responses of these two spaces for the centre voxel where the three line segments meet. In Fig. 2.3(e-f) we show two polar Voronoi plots of the angular response of the two orientation spaces. A Voronoi cell is defined as the set of orientations closer to $\boldsymbol{\phi}_i$ than to all other $\boldsymbol{\phi}_{j\neq i}$. In the plot each Voronoi cell has a gray value corresponding to the height of the orientation space response. In the plot, θ is the radial and ϕ the angular coordinate. Orientation (ϕ, θ)=(0,0) can

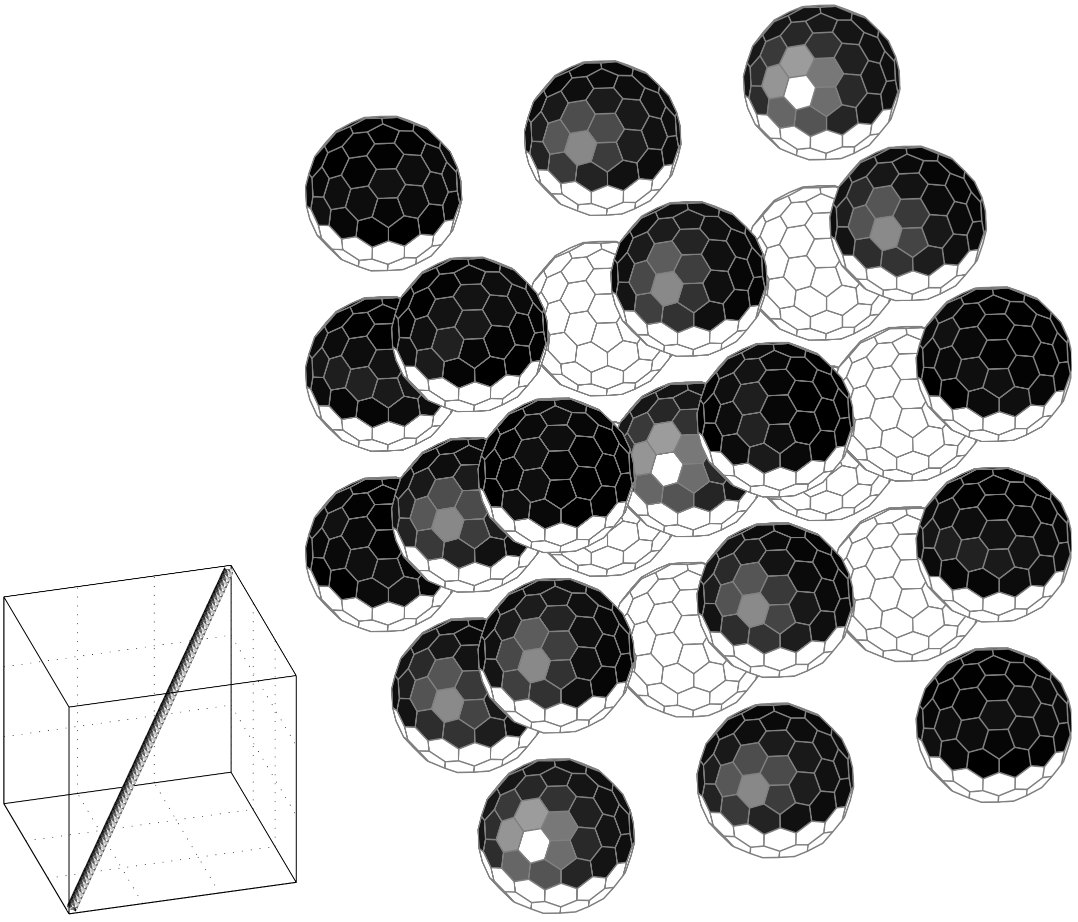


Figure 2.4: (a) isosurface of a Gaussian line. (b) the corresponding 5D orientation space of the 27 voxels of the image in (a).

be found in the centre of the plot and orientations with $\theta = \pi/2$ can be found on the inner solid circle. In Fig. 2.3(e) we can see three peaks. But the two peaks centered around (0,$\pi/2$) and (0,$3/2\pi$) belong to one and the same orientation since a line through the centre of a sphere yields two crossings at opposite sides. More interesting is the other peak with the elliptical shape. This shape indicates that the underlying structure(s) are not resolved, as predicted by the angular resolution criterion. In Fig. 2.3(f) the number of sampling orientations is increased from 81 to 181. As can be seen the responses of the two lines are now nicely resolved. Therefore, only the orientation space with 181 sampling points can be used to segment the fork image.

Now let us look at another visualisation method for a 5D orientation space. In Fig. 2.4(a) we show an image with a line through its centre. In Fig. 2.4(b) we plot the orientation space (46 sample orientations) of the 27 voxels in the centre of the image. On the individual spheres the Voronoi cells are plotted in the same way as for the polar Voronoi plot. With orientation $\{\phi, \theta\} = \{0, 0\}$ on top of the spheres and all spheres rotated through an angle π around the ϕ-axis. In the image we see that the response is localised and drops off very quickly with the distance to the line. It is actually possible to prove it has a Gaussian profile.

2.5 Conclusions

We have extended the principle of orientation space to 3D images by designing orientation selective line and plane/edge filters. After an elaborate search for sampling grids on a sphere we selected a multi-scale grid based on the icosahedron which allows easy addressing. We presented methods to visualise the resulting 5D structure. The method can deal with multiple intersecting oriented structures and contains local orientation information. In the future we will try to produce an orientation space formulation which is sparse and local. This to limit the computational burden and memory consumption. Further we will investigate some interesting applications of the orientation space approach.

A crossing detector based on the structure tensor

3

abstract*

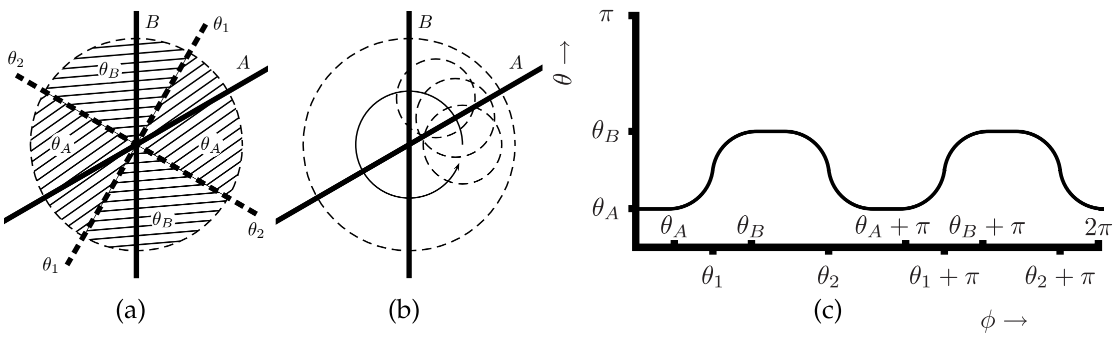
A new crossing detector is presented which also permits orientation estimation of the underlying structures. The method relies on well established tools such as the structure tensor, the double angle mapping and descriptors for second order variations. The performance of our joint crossing detector and multi-orientation estimator is relatively independent of the angular separation of the underlying unimodal structures.

3.1 Introduction

The structure tensor [8] and its nonlinear variations [16] yield a reliable estimate of orientation on unimodal structures. It fails where unimodal structures overlap (or cross). In this paper we present a method based on the structure tensor to divide the image around crossings in unimodal regions. Using the 4-fold symmetry of the orientation map at line crossings (or saddle points in checkerboard patterns) we are able to achieve a high response independent of the angular separations of the underlying lines. This in contrast to e.g. the Harris Stephens crossing detector [58] and variations thereof [114, 135, 50, 42, 128, 70] for which the response drops significantly with decreasing angular separation. Our new method is reasonably fast, has a good angular selectivity and yields good localisation. This is particularly important for camera calibration in which the crossings of checkerboard patterns (or other fiducials) need to be located with sub-pixel accuracy in many different poses. Another key application in molecular biology requires the detection and characterisation of overlapping bio-polymers such as DNA strands deposited on a surface for AFM or TEM imaging.

---

[*] The content of this chapter has been published in: F. G. A. Faas and L. J. van Vliet. A crossing detector based on the structure tensor. In J. Blanc-Talon, W. Philips, D. C. Popescu, and P. Scheunders, editors, *ACIVS 2007, Advanced Concepts for Intelligent Vision Systems, 9th International Conference, (Delft, The Netherlands)*, LNCS 4678, pages 212–220, August 28-31 2007 [34].



(a) (b) (c)

Figure 3.1: (a) Sketch of crossing lines A and B with respectively orientations θ_A and θ_B. The dashed lines denote the major axes of the detected saddle point at the crossing with respective orientations θ_1 and θ_2. The dashed regions denote the areas in which the measured orientation by the GST in the circular region is approximately constant. In sketch (b) the same crossing is denoted. The circular arrow denotes the track along which the orientation response of the GST is sketched in subfigure (c)(phase wrapping is assumed absent).

3.2 Method

The key observation to our method is the following. Applying the gradient structure tensor [8, 58](GST) to a crossing of linear structures results in an orientation pattern with a saddle point structure, i.e. regions of uniform orientation bounded by the bisectors of the underlying crossing (Fig. 3.1). Hence detection of these saddle points will yield a crossing detector. After the orientation of these saddle points is determined one can divide the local neighbourhood of a crossing in four regions, i.e. wedges with an opening angle of $\frac{\pi}{2}$ radians. The antipodal wedges can form two bow ties, see the regions in Fig. 3.1(a) labelled with respectively θ_A and θ_B for lines A and B. Applying the gradient structure tensor to these regions separately, either to the bow tie or to the wedges separately, yields a reliable local orientation estimate for each arm or line of the crossing separately. As the location of the crossing is already detected, the orientation estimate of the four wedges can be converted to direction estimates pointing away from the crossing's centre. In this section we will briefly describe the four steps of our method.

Transform crossings into saddle shaped structures:

In the first step of our algorithm we determine the local orientation by means of the GST. The GST, $\overline{T(I)}$, is the averaged dyadic product of the gradient field ∇I of image I, in which the overhead bar denotes local averaging.

$$\overline{T(I)} = \overline{\nabla I \nabla I^{\mathrm{T}}} \quad \text{with} \quad \nabla I = [I_x, I_y]^{\mathrm{T}}. \tag{3.1}$$

In this tensor representation two antipodal vectors are mapped on top of eachother. Where the antipodal vectors cancel out in an averaging step, the corresponding tensor representations reinforce eachother. Now the directional gradient power is maximised for angle θ

$$\tan 2\theta = 2\overline{I_x I_y} \,/\, (\overline{I_x^2} - \overline{I_y^2}) \tag{3.2}$$

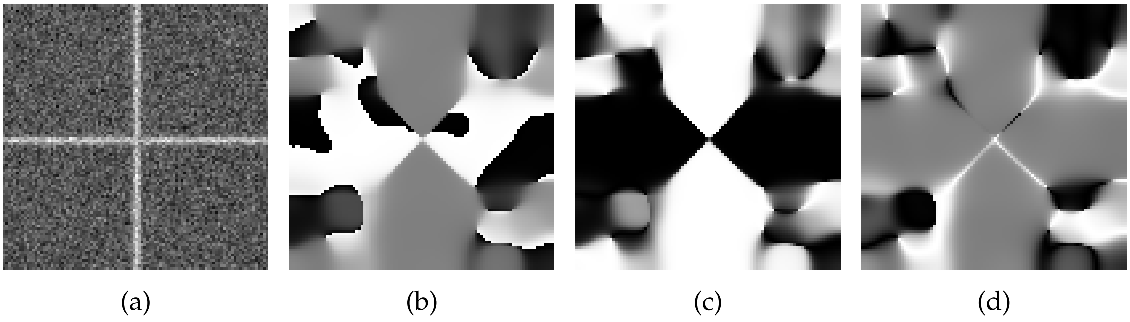


Figure 3.2: (a) Synthetic image of a cross. (b) Orientation determined by GST clearly showing phase wrapping (BW transitions). (c-d) The double angle mapping of (b) with respectively in (c-d) the cosine and sin of the double angle of (b).

where θ denotes the orientation of the gradients of the unimodal structure. This corresponds to the orientation of the eigenvector belonging to the largest eigenvalue. In Fig. 3.1(a) the saddle point structure of the orientation field of the GST is sketched. This saddle point structure is caused by the averaging nature of the GST which treats the local neighbourhood as a single structure, i.e. when one of the arms is dominant in the analysis window it will dominate the orientation result as well. Only when both lines are visible an averaged orientation will be obtained, see Fig. 3.1(b-c). Due to this averaging property of the GST the orientation field will look like a Voronoi tessellation of the underlying structures, i.e. in case of crossing lines a cross is formed by the internal bisectors of the lines, see Fig. 3.1(a). Although the GST gives an excellent characterisation of the local orientation, the angle representation of Eq. 3.2 suffers from phase wrapping, i.e. the resulting orientation is modulo π radians. This causes large jumps in the orientation image where the angle jumps from 0 to π radians while in reality these orientations are identical. In Fig. 3.2(a-b) we show respectively a synthetic crossing and the orientation estimate by means of the GST. Where the latter clearly shows phase wrapping events.

To solve the phase jumps caused by phase wrapping we apply a double angle mapping to the measured orientation.

$$\theta \rightarrow (\cos 2\theta, \sin 2\theta) \tag{3.3}$$

Note that the double angle is closely related to the GST. As shown in [55] this mapping preserves the angular metric, gives a continuous mapping and preserves the local structure. In Fig. 3.2(c-d) the double angle representation is shown for the image in Fig. 3.2(b). It clearly shows that the phase wrapping events, BW transitions, in Fig. 3.2(b), are absent after the double angle mapping, Fig. 3.2(c-d).

Generate candidate crossings from second order shape descriptors:

The phase unwrapped orientation gives rise to a saddle structure. This structure is more pronounced for large angles of separation, i.e. for lines crossing at an angle of $\frac{\pi}{2}$ radians it is maximised. Therefore a saddle point candidate generator is needed which separates the $magnitude$ of the saddle point from the $shape\ descriptor$ which characterises the structure type. To this end we explore the second order structure

[26]. A structure vector ***f*** is presented with three components to describe the second order structure based on the Hessian matrix. These three components are respectively the angle β which denotes the orientation of the structure, κ which denotes the structure type and f which denotes the structure strength. These descriptors are based on the spherical harmonics which constitute an orthonormal basis in contrast to the second order derivatives which are not independent. However, the second order spherical harmonics J_{ij} can be expressed in terms of the second order derivatives J_{ab} as follows

$$\begin{pmatrix} J_{20} \\ J_{21} \\ J_{22} \end{pmatrix} = \frac{1}{\sqrt{3}} \begin{pmatrix} & (J_{xx} + J_{yy}) \\ \sqrt{2} & (J_{xx} - J_{yy}) \\ \sqrt{8} & J_{xy} \end{pmatrix} \tag{3.4}$$

Now we can express the structure vector ***f*** in terms of the spherical harmonics for image J as,

$$\boldsymbol{f}(\mathrm{J}) = \begin{pmatrix} f \\ \beta \\ \kappa \end{pmatrix} = \begin{pmatrix} |\,(J_{20}, J_{21}, J_{22})\,| \\ \arg(J_{21}, J_{22}) \\ \arctan(J_{20}/\sqrt{J_{21}^2 + J_{22}^2}\,) \end{pmatrix} \tag{3.5}$$

For $|\kappa| = \frac{\pi}{2}$ the structure can be described as a blob structure, for $|\kappa| = \frac{\pi}{6}$ as ridges/valleys and for $|\kappa| = \frac{\pi}{3}$ we have the pure second order derivatives. The pure saddle structure is located at $|\kappa| = 0$.

The double angle representation results in two κ images, i.e. one for the sine and cosine term. These images are combined in one structure descriptor κ' based on the corresponding structure strength, i.e.

$$\kappa'(\theta) = \begin{cases} \kappa(\cos 2\theta) & \text{if} \quad f(\cos 2\theta) > f(\sin 2\theta) \\ \kappa(\sin 2\theta) & \text{elsewhere} \end{cases} \tag{3.6}$$

To detect candidate saddle points we apply a threshold to the $|\kappa'|$ image,

$$Saddle(\theta) = \begin{cases} 1 & \text{if} \quad |\kappa'| \le \kappa_{th} \quad \text{with} \quad \kappa_{th} = \frac{\pi}{12} \\ 0 & \text{else} \end{cases} \tag{3.7}$$

where κ_{th} is chosen as the middle value between the pure saddle point at $|\kappa'| = 0$ and the line structures at $|\kappa'| = \frac{\pi}{6}$.

Detect crossings using a second order magnitude measure:

After we generated the candidate saddle points we want to assign a magnitude measure to each candidate based on the structure strength to confine the candidates to regions where structure is present, i.e. noise can also give rise to saddle points on a small scale. As $f(\theta)$ based terms are dependent on the angular separation of the crossing, another energy measure is needed to reduce the angular dependency of the detector. The measure of our choice is the curvature-signed second order strength in I,

$$E(I) = \mathrm{sign}(\kappa(I))\,|f(I)| \tag{3.8}$$

where the sign term is introduced to be able to distinguish between the crest of a line and its edges, i.e. on a ridge the curvature is positive but on the flanks the curvature is negative. Thresholding the energy image yields the candidate regions based on the structure strength in I,

$$Energy = \begin{cases} 1 & \text{if} \quad E \gtrless E_{th} \quad \text{with} \quad E_{th} = \text{threshold}(\{E(I) | E \gtrless 0\}) \\ 0 & \text{else} \end{cases} \tag{3.9}$$

where the comparison direction depends on the structure of interest, i.e. black lines on a white background or vice versa. E_{th} is determined by an isodata threshold on respectively the positive or negative data in $E(I)$, the threshold type can of course be adapted to a particular problem. Now we combine the $Energy$ and $Saddle$ images by an AND operation. Furthermore, to remove spurious detections, we require the detected regions to be larger than S_A pixels.

$$Detector = \{x | x \in Saddle \wedge x \in Energy \wedge Area(x) > S_A\} \tag{3.10}$$

where the area S_A is defined as the minimum cross section of two lines of width w intersecting under an angle ϕ, i.e. $S_A = w^2$. Of course, the line width of the line is a combination of the true width of the line, the PSF of the imaging device and size of the derivative kernels and as such has to be set to a suitable value for the problem at hand.

Analyse orientation of lines composing the crossing:

The algorithm continues with the analysis of the orientation of the lines from which the crossing is composed. Therefore, first the centre of gravity is determined for each connected region in the detector image, which serves as location of the detected crossing. Further analsis is performed with these points as point of origin. The value of β at these points now gives the orientation of the saddle points (the β responses on the double angle representation are combined in a similar fashion as those for κ in Eq. 3.6). At these points the gradient information is analysed by means of the GST in the bow tie shaped region. The bow tie is constructed by the major axes of the saddle point, i.e. given by lines through the local point of origin with orientation β and $\beta + \frac{\pi}{2}$. The eigenvector belonging to the largest eigenvalue of the GST for each bow tie gives the orientation of the underlying structure.

The size of the gradients (Gaussian derivatives of scale σ_g) must be small to avoid unnecessary signal suppression. The size of the tensor smoothing (Gaussian filter of scale σ_t) is usually three to five times larger than the gradient size. The size of the second derivatives is set identical to the size of tensor smoothing.

3.3 Results

First we test the algorithm on synthetic data, i.e. crossing lines with an angular separation between 0 and $\frac{\pi}{2}$ radians, see e.g. Fig. 3.2(a). In Fig. 3.3 we show the results for lines with a Gaussian line profile of $\sigma_{line} = 1$ and a SNR of respectively 10 dB and 25dB after addition of Gaussian noise. The signal to noise ratio

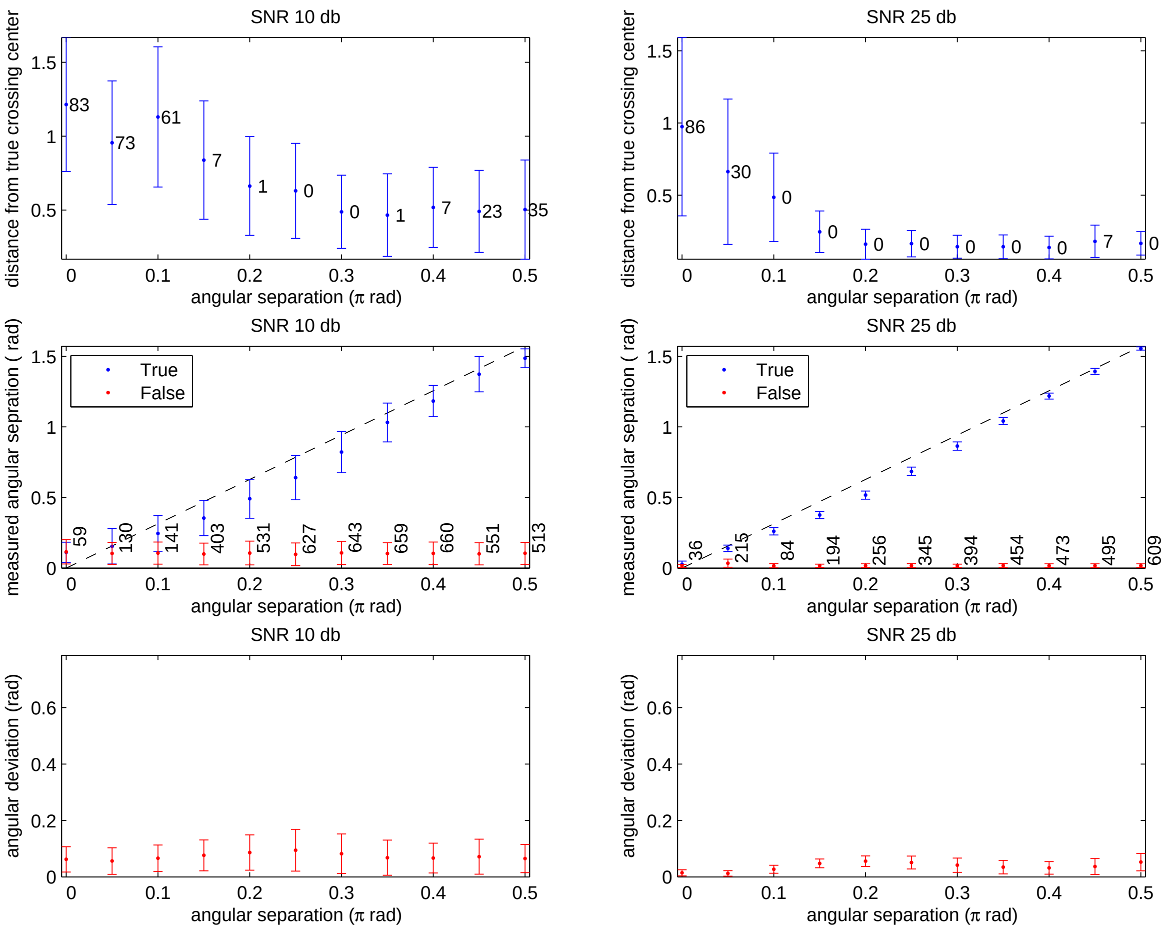
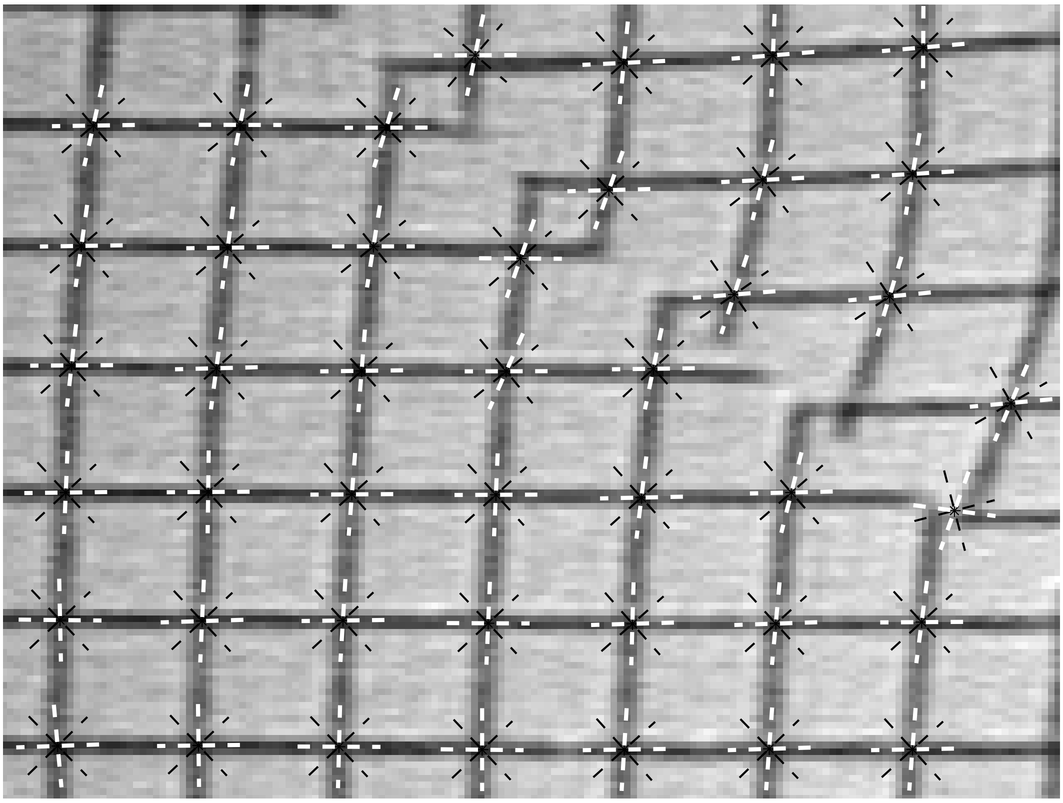
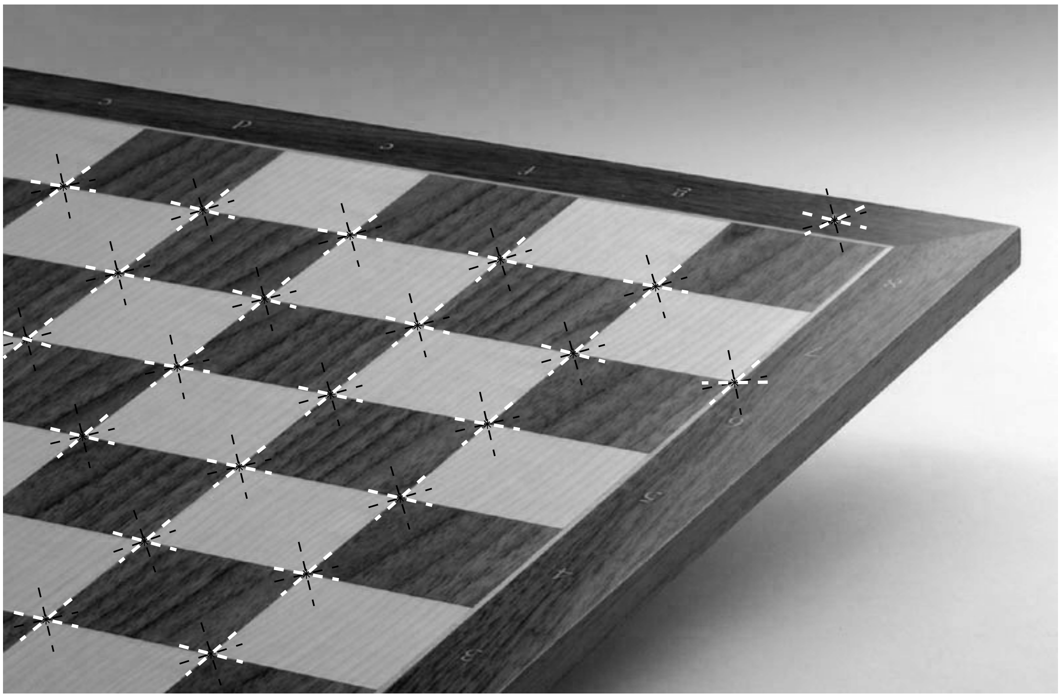


Figure 3.3: Top row figures show the distance between the true crossing centre and the measured centre as a function of the angular separation of the lines for true positives. The middle row shows the measured angular separation of the true crossings (in blue) and the false positives (in red). The plots on the bottom row show the angular deviation of the measured orientations from the true orientations of the lines for true positives. The left and right column show the results for 10dB and 25 dB respectively. For each separation angle 100 realizations were made.

is defined as SNR$=20\log\left(contrast/\sigma_{noise}\right)$. For each angular separation 100 realizations are obtained with randomly selected sub-pixel position and orientation of the structures. All derivatives and averages are computed with Gaussian kernels ($\sigma_g = 1, \sigma_t = \sigma_s = 4$). Note that for both noise levels the same settings were used. The region in which the orientations of the lines were measured, complies to the size of the tensor smoothing. Further, the analysis window is set to a region within 100 pixels from the crossing centre. Keep this in mind as the number of false positives are expected to scale linearly with this value. On the top row of Fig. 3.3 the distance from the true centre is depicted, where the numbers denote the number of false negatives. True positives are detections closest to the true centre and at a maximum distance of 2 pixels. All other detections are marked as false positives. The high number of false negatives for small angles is attributed to the the fact that a crossing resembles more and more a line with decreasing angular separation resulting in a poor localisation. The increase in the number of false negatives for large separations in the high noise



(a)



(b)

Figure 3.4: (a) Image of a deformed miniaturised clay dike model with a superimposed grid. Courtesy of GeoDelft, The Netherlands. (b) Checker board image. In both images the white dashed lines denote the orientation estimates of the detected crossings while the black dashed lines show the major axes of the saddle point regions.

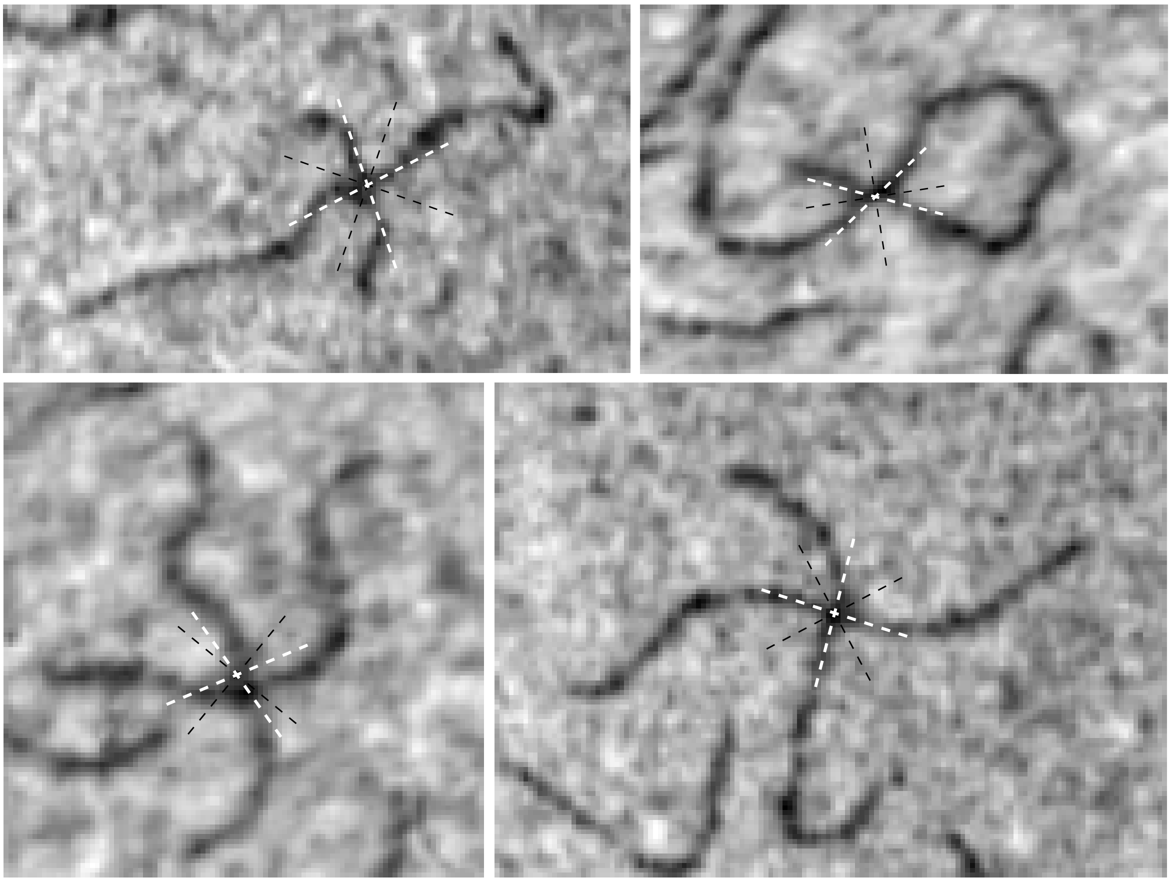


Figure 3.5: DNA molecules labelled with uranyl acetate and visualised by transmission electron microscopy. The images are kindly provided by Dr. D. Cherny, PhD, Dr.Sc. The examples show (self) crossing DNA strands, the white dashed lines show the orientation estimates of the detected crossings while the black dashed lines show the major axes of the saddle point regions.

realizations is not fully understood, but can be lowered in exchange for more false positives with small separation angles. On the middle row of Fig. 3.3 the measured angular separation is plotted as a function of the true angular separation, where the numbers denote the number of false positives. The plots clearly show that the false positives can be easily separated from the true positives for separation angles larger than $\approx \frac{\pi}{10}$, and even smaller for the low noise case. The figures on the right denote the error in the orientation estimation of the crossing lines.

In Fig. 3.5 and 3.4 we show some examples on real data. The images represent respectively DNA strands, a deformed clay dike model and a checker board. For these images $S_A = \sqrt{(1+\sigma_t^2)}$, i.e. the width of the detected lines is put to 1. The settings for the first order derivatives, tensor smoothing/second derivatives and the cutoff radius of the wedges are respectively $(\sigma_g, \sigma_t) = (2, 10)$,$(1, 4)$ and $(1, 6)$ for the clay dike, checker board and DNA images.The clay dike image is produced by line scanning and suffers from striping. To overcome this problem the tensor smoothing is set to a relative high value. In all three images the black dashed lines denote the major axes of the saddle points whereas the white dashed lines denote the measured orientations of the underlying lines.

3.4 Conclusions

The presented crossing detector is relatively insensitive to the angular separation of constituent lines/edges. False positives can easily be removed by setting a simple threshold on the angular separation. The detector also allows for an accurate orientation estimation of the underlying structures and performs well on noisy data. We believe this can be a good tool for camera calibration on checkerboard images due to its independence of the angular separation between the linear structures (pose independence). Further it can be used for the analysis of (self)overlapping line-like objects. The low number of parameters can be adjusted easily to the problem at hand where the values are intuitive to determine. For the first order derivatives we like to keep the footprints as small as possible. The tensor and second order footprints can be kept at the same value where the value is dependent on the spatial separation of crossings as well as the noise properties of the image at hand. The same is true for the final orientation measurements of the underlying structures. The size of the bow ties are $\sqrt{2}$ time the size of the tensor smoothing.

Junction detection and multi-orientation analysis using streamlines

4

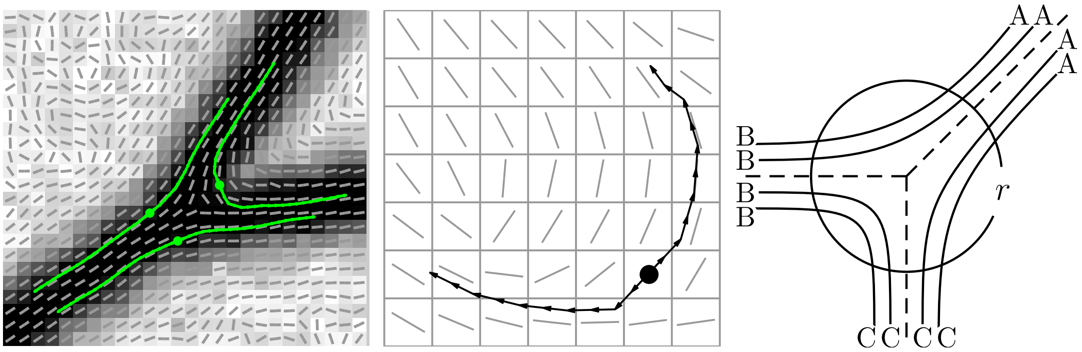
abstract[*]
We present a novel method to detect multimodal regions composed of linear structures and measure the orientations in these regions, i.e. at line X-sings, T-junctions and Y-forks. In such complex regions an orientation detector should unmix the contributions of the unimodal structures. In our approach we first define a (streamline) divergence metric and apply it to our streamline field to detect junctions. After this step we select all streamlines that intersect a circle of radius r around the junction twice, cluster the intersection points and compute the direction per cluster. This yields a multimodal descriptor of the local orientations in the vicinity of the detected junctions. The method is suited for global analysis and has moderate memory requirements.

4.1 Introduction

Several well known tools exist for the detection, localisation and characterisation of unimodal linear structures based on the structure tensor and the Hessian matrix. For multimodal structures the detection can still rely on the structure tensor [55], as demonstrated by the Harris-Stephens corner detector [58]. However these methods cannot unmix the responses from the composing structures. Most detectors have a response which decreases strongly with decreasing angular separation between the composing structures. The filterbank approach [46, 52, 35] is a well known method for the analysis of multimodal regions. However, its excellent angular selectivity is combined with a poor localisation.

---

[*] The content of this chapter has been published in: F. G. A. Faas and L. J. van Vliet. Junction detection and multi-orientation analysis using streamlines. In W. G. Kropatsch, M. Kampel, and A. Hanbury, editors, *CAIP 2007, Computer Analysis of Images and Patterns, 12th International Conference, (Vienna, Austria)*, pages 718–725, August 27-29 2007 [39].



(a) (b) (c)

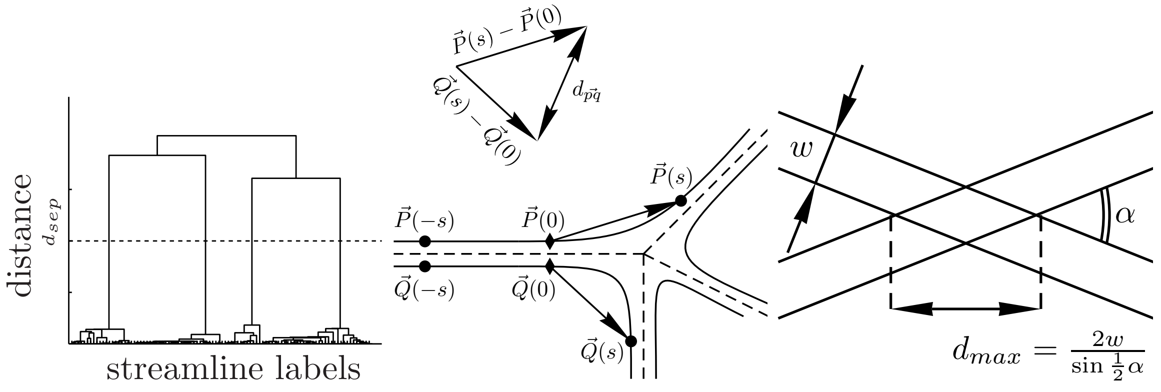
Figure 4.1: (a) Artificial Y-junction with orientation field (gray overlay) and three streamlines (green). (b) Orientation field with a streamline starting from the black dot in the center. (c) Sketch of a Y-junction (dashed lines) with some streamlines around the junction center. The circle labeled with r denotes the distance from the center at which the orientation field is sampled and the labeling of the streamlines is performed. The letters denote the (double) labeling of the streamlines.

The key observation to our method is the following: any X-sing, T-junction or Y-fork is composed of a few unimodal structures (line or edge segments). At X,T,Y-transitions a single measure for orientation, e.g. by means of the structure tensor, yields a weighted sum of orientations of the constituent elements. Moving away from the centre of the junction toward one of the unimodal structures, the measure will approach that of a unimodal structure.

In this paper we present a streamline based method which connects the unimodal regions to multimodal regions, e.g. the arms of a junction to the junction itself. The streamlines are constructed from a vector field that represents the local structure. This method allows us to accurately locate junctions and crossings and at the same time measures the attributes of the underlying unimodal structures. The method offers a good angular selectivity within a relatively small analysis window.

4.2 Method

Our method comprises five steps. The first step is to find a suitable vector field which describes the local structure in a consistent way, see Fig. 4.1(a-b). Second, this vector field is used to calculate streamlines at each pixel position. Third, a streamline divergence metric is introduced which yields a high value for multimodal structures and a low value on simple linear structures. Fourth, a suitable threshold is applied to detect the multimodal structures. After merging fragmented responses, the centre point of the junction is determined. Fifth, we procede with the analysis of the streamlines in a region around the junction centre, see Fig. 4.1(c). To that end we define a circle with radius r centered at the junction. A suitable streamline should cross this circle twice, i.e. once for each of the unimodal structures it connects. The points where the streamlines intersect with the circle defines the new endpoints. The set of all endpoints will give rise to clusters, i.e. one cluster for each unimodal structure in the junction. These clusters are analysed and each streamline is assigned two



(a) (b) (c)

Figure 4.2: (a) Typical dendrogram for a X-sing, depicting the cluster distance measure (vertical) composed of individual streamlines (horizontal). (b) Sketch of the distance measure $d_{\boldsymbol{pq}}$ between the streamlines $\boldsymbol{P}$ and $\boldsymbol{Q}$ at a distance s from the respective starting points $\boldsymbol{P}(0)$ and $\boldsymbol{Q}(0)$. (c) Sketch of the intrinsic scale (here denoted by d_{max}) of the intersection of two lines of width w.

labels, one for each unimodal structure it belongs to. For each cluster we compute the average direction over the endpoints of the streamlines that belong to the same cluster.

step one: vector field

A suitable vector field in this framework follows the underlying local structure in the image and is smooth and continuous. The vector field should be well defined on both even and odd structures, i.e. respectively along the bottom of the valley and along the edge of linear structures. We assume, without loss of generality, a white background and dark foreground throughout this paper.
The vector field in this paper is created by means of a set of rotated quadrature filters which are combined into a tensor representation. An eigensystem analysis gives the vector field. The tensor is constructed from quadrature filters as described by [72]. The quadrature filter is constructed by means of the generalised Hilbert transform

$$F_i = G(|\boldsymbol{u}|, \sigma_F)\, H(\boldsymbol{u} \cdot \boldsymbol{n}_i)\, (\boldsymbol{u} \cdot \boldsymbol{n}_i)^2 \tag{4.1}$$

with G an isotropic Gaussian transfer function, H the Heaviside function and the last factor a quadratic cosine term to select the radial polynomial as well as the angular response. The orientation vectors $\boldsymbol{n_i}$ are defined as $\boldsymbol{n_i} = [cos(\frac{\pi}{3}i), sin(\frac{\pi}{3}i)]^T$ with $i = 0, 1, 2$. The 2D tensor T is defined as

$$\mathrm{T} = \sum_{i=1,2,3} |q_i|(n_i n_i^T - \mathrm{I}) \tag{4.2}$$

with q_i the response of the i^{th} quadrature filter in the spatial domain. Now the vector field is given by the eigenvector belonging to the smallest eigenvalue.

step two: streamlines

A streamline is a line which is everywhere tangent to the local flow field. Mathematically this can be stated as

$$\frac{d\boldsymbol{x}}{ds} \propto \boldsymbol{u}(\boldsymbol{x}) \; and \; \boldsymbol{x}(s_0) = \boldsymbol{x}_0 \tag{4.3}$$

with $\boldsymbol{u}$ the vector field and $\boldsymbol{x}(s)$ the parameterised streamline. Further, point $\boldsymbol{x}_0$ denotes a point on the streamline, i.e. the point of origin. In here a streamline is defined as a curve which is tangent to the local structure. As even structures have no direction, the flow vector is equally well described by two antipodal vectors, creating possible phase jumps between neighboring points. To solve this phase jump problem locally , i.e. at position $\boldsymbol{x}(s)$ on the streamline, the flow field is given by one of the two antipodal vectors which produces the smallest angle with respect to the tangent vector of the streamline at $\boldsymbol{x}(s)$. As such the flow field is defined with respect to a local point of origin and the propagation direction of the streamline. Further, the start direction, i.e. $\boldsymbol{u}(s_o)$ of the streamline is ambiguous as there is no history to determine the tangent vector. This is not a problem as the curve of interest is centered around the point $\boldsymbol{x}_0$ and extends along the structure in both directions for a distance l resulting in a streamline of length $2l$ with parameter s running from $-l$ to l along the curve, i.e. from one end to the other end. Here s is the position along the contour measured from the centre point. The streamlines are implemented by means of a first order method.

step three: streamline divergence metric

Two streamlines originating from points close together, on say one arm of the junction, can end up in two different arms separated by a significant distance, $d_{\boldsymbol{pq}}$, see Fig. 4.2(b). Our junction detector is based on this observation. For each pixel the streamlines in a neighbourhood are analysed and the maximum separation between the reference streamline and the streamlines in the neighbourhood are determined. We compute the distance between two streamlines at a distance l_d (measured along the streamline) away from the pixel to be inspected. Summing over the line distances in the neighbourhood yields a high value in the proximity of a junction.
The streamline, $\boldsymbol{P}$, through point $\boldsymbol{p}$ has to be compared to the test streamline, $\boldsymbol{Q}$, through point $\boldsymbol{q}$, where $\boldsymbol{q}$ lies in the neighbourhood of $\boldsymbol{p}$. The first step in computing the maximum separation between two streamlines is to align them with respect to the streamline at the point of interest $\boldsymbol{p}$. This is necessary because the 'positive' direction of the streamline is ambiguous along line structures in the image. After alignment the two streamlines in negative direction converge and in the positive direction diverge. Now the distance between the two streams $\boldsymbol{P}$ and $\boldsymbol{Q}$ is defined as the distance between the points on the streams with a distance l_d to their respective origin measured in the aligned positive direction;

$$d_{\boldsymbol{pq}} = |(\boldsymbol{P}(l_d) - \boldsymbol{Q}(l_d)) - (\boldsymbol{P}(0) - \boldsymbol{Q}(0))| \tag{4.4}$$

where the second term is the translation vector between the streamline centres, see Fig. 4.2(b). Now we sum the distances between the streamline at $\boldsymbol{p}$ and those in the local neighbourhood N to get a 'streamline divergence' measure $d_{\boldsymbol{p}}$ at $\boldsymbol{p}$;

$$d_{\boldsymbol{p}} = \sum_{\forall \boldsymbol{q} \in N} d_{\boldsymbol{pq}} \tag{4.5}$$

in which the size of neighbourhood N reflects the intrinsic scale of the constituting unimodal structures. The intrinsic scale is defined by the width of the line segments or the length of the edge slope. It is by definition larger or equal to the support of the overall Point Spread Function of optics and filters and can be measured by the method of Dijk [29].

step four: detection of junctions

The aforementioned 'streamline divergence' measure also gives reasonably high responses in the (noisy) background. We suppress these points by multiplying with the certainty of the Hessian matrix which is close to zero in the background. A measure based on the tensor T is rejected as its response is not selective and not localised enough due to the quadrature filters. Therefore we use the more localised certainty measure based on the Hessian matrix as presented in [26] to select our regions of interest, basically the valley regions. This certainty measure can be stated as

$$c = |f_{20}, f_{21}, f_{22}| = \sqrt{f_{xx}^2 + f_{yy}^2 - \frac{2}{3} f_{xx} f_{yy} + \frac{8}{3} f_{xy}^2} \tag{4.6}$$

where f_{2i}, with $i \in \{0, 1, 2\}$, denote the spherical harmonics of second order and $\{f_{xx}, f_{xy}, f_{yy}\}$ the second order spatial derivatives. This certainty deviates from the norm of second order derivatives, i.e $c \neq |f_{xx}, f_{xy}, f_{yy}|$, as it corrects for the fact that the second order spatial derivatives do not form an orthogonal basis [26] while the spherical harmonics of the second order do. Now an isodata (two-means) threshold [107] on the measure c is performed resulting in regions with a pronounced second order structure. As we are only interested in the valleys, the following restriction is introduced $|\lambda_1| > |\lambda_2| \wedge \lambda_1 > 0$ where λ_i is the i^{th} eigenvalue of the Hessian matrix with $\lambda_1 > \lambda_2$. The regions for which one of the restrictions fails is put to zero in the response image. Of course, other nonlinear weightings can also be applied to suppress the background.

Now we can apply a simple threshold to the cleaned response image (just above the distinct background peak of the corresponding histogram) to get points where the streamlines show a high degree of divergence. Right at the junctions eddies may occur due to a lack of translational invariance. Hence we get a relatively low response at the centres of these junctions. However close to these vortexes the divergence of the streamlines is very high. Therefore we include a step in which we merge the divergence responses in a neighbourhood equal to the intrinsic scale of the underlying structure. Now we have identified the junctions we use the centre of gravity of those regions as the position of the junctions. The localisation can be further refined by finding the points of minimum distance to the found centre point on all streamlines in the junction region. From these points a new centre of gravity can be calculated and if necessary this can be iterated. In our implementation the last step of refining the localisation is not performed as the localisation was good enough for our aim of demonstrating the algorithm.

step five: direction estimation

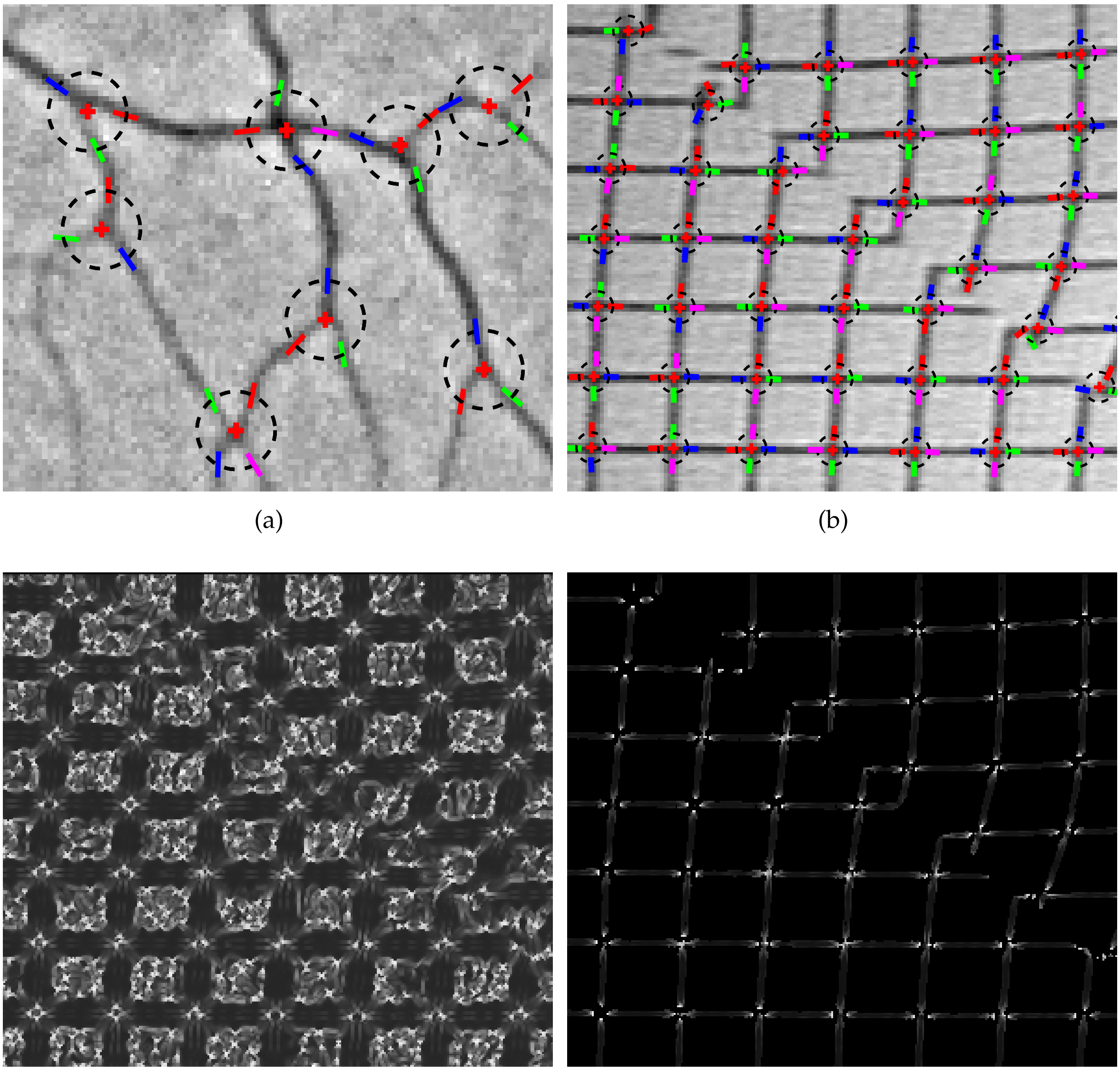
In the metric subsection we aligned streamlines on a pair by pair basis. In this paragraph we will cluster them such that they are labelled with the two labels of the two unimodal structures they connect. This is done by a simple hierarchical clustering method and applying a threshold at d_{sep} to the dendrogram, see Fig. 4.2(a). In Fig. 4.1(c) a junction is shown with a number of streams each with two labels, i.e. A, B or C. We define a circle with radius r centered around the junction centre. Now from all streamlines the intersections with this circle are determined, i.e. each streamline should cross this circle twice. Based on these points we will label our streamlines. r should be chosen such that the circle is larger than the size of the mixed zone and on the other hand as small as possible to avoid mixing with neighboring junctions. The longest 'diameter', d_{max}, of the mixed zone depends on the intrinsic scale of the constituent unimodal structures and the smallest angle α between them, see Fig. 4.2(c). Note that we have directional information due to the fact that the orientation "vectors" can be oriented such that they point away from the junction centre. These vectors can therefore be averaged in each cluster i.e. no structure tensor is necessary as all vectors are mapped in the same direction, i.e. away from the junction. This average is used as direction estimate. The directional information of the streamlines is sampled at r, see Fig. 4.1(c).

4.3 Results

Fig. 4.3(a) and (b) show respectively a small part of the vascular system in the retina of an eye and a deformed miniaturised claydike model with a superimposed grid. The overlays in both images show the locations of the detected crossings (red crosses), the circles on which the directions are measured and the measured orientations (colored lines). The minimum distance between clusters is set to $d_{sep} = 1$ which means that clusters separated by a distance smaller than d_{sep} in the dendrogram are merged. Further we set r and l_d to identical values, i.e. $r = l_d = 7$ and $r = l_d = 8$ for respectively the retina and claydike image. These values roughly correspond to the scale of the crossings in the images. Further at a distance l_d from the junction centre the streamlines are in general parallel to the unimodal structures. As can be seen all crossings are detected. The direction estimate for one of the arms of left middle junction in the retina image is a bit off. This is a consequence of the very low contrast, i.e. the influence of the noise on the vector field is more severe if the contrast is low. Fig. 4.3(c-d) show respectively the 'streamline divergence' before and after background suppression using the second order certainty measure.

4.4 Discussion and conclusions

Preliminary testing of our method shows excellent detection and characterisation for a modest window size. The junction detection is dependent on the local structure and as such relatively independent of the local contrast. Furthermore, the angular selectivity can be increased by increasing the labelling distance r. Another nice aspect of the method is that it automatically selects a mode, i.e. it selects the number



(a) (b)

(c) (d)

Figure 4.3: (a) Blowup of a retina image (courtesy National Eye Institute, U.S. National Institute of Health). (b) Image of a deformed miniaturized claydike model with a superimposed grid. Courtesy of GeoDelft, The Netherlands. In overlay: the dashed circles denote the analysis window and the colored lines at the circle denote the measured orientations at their intersection with the circle. With $r = l_d = 7$ and $r = l_d = 8$ or respectively image a and b. (c) The distance metric of the image in b. (d) The distance metric for the image b after setting the points where the Hessian is uncertain to zero.

of unimodal structures at the junction.

The implementation used can easily be improved e.g. by means of higher order streamline methods. This should decrease the influence of noise and should refine the streamlines. Furthermore one could think of averaging the orientation information over a larger area to obtain a more robust estimate, .i.e. at this moment the streams are only sampled were they cross a circle of radius r which makes the orientation susceptible to noise.

We state that the labelling of the streamlines can be performed at approximately the same distance from the junction centre as the detection of the junctions, i.e. $l_d \approx r$, because both parameters are directly related to the intrinsic size of the junctions. As shown in Fig. 4.2(c) we can relate the minimum angle of separation α to the width w of the unimodal structures. As such as a rule of thumb we state that $r \leq \frac{2w}{\sin \frac{1}{2}\alpha}$ which ensures that the entire mixed zone is enclosed. As such the parameters are basically reduced to an angular resolution parameter α and the width of the unimodal structures.

Optionally one could perform the clustering and the direction measurements at different positions along the streamlines. As the intrinsic scale of junctions can differ over the image one could make r and l_d position dependent [50]. This could also minimise interaction of structures in close proximity of each other. After inspection of the individual steps of the algorithm we estimate that the time complexity of the algorithm is approximately a factor of 10 higher than that of a eigensystem analysis of the gradient structure tensor. Further the data complexity is low and only widely accepted tools are used.

Multi orientation analysis by decomposing the structure tensor and clustering

5

abstract[*]

The structure tensor yields an excellent characterisation of the local dimensionality and the corresponding orientation for simple neighbourhoods, i.e. neighbourhoods exhibiting a single orientation. We show that we can disentangle crossing structures if the tensor scale is much larger than the gradient scale. Mapping the gradient vectors to a continuous orientation representation yields a $\frac{1}{2}D(D+1)$-dimensional feature vector per pixel. Clustering of the vectors in this new space allows identification of multiple orientations. Each cluster of gradient vectors can be separately analysed using the structure tensor approach. Proper clustering yields an unbiased estimate of the underlying orientations.

5.1 Introduction

The structure tensor is a powerful tool for the analysis and characterisation of structures in multi-dimensional images. [58] and [8] developed the structure tensor independently for solving two completely different tasks in image processing and computer vision. Without formulating a tensor and before it was discovered, [67] presented an algorithm for estimating the average local orientation in 2D images of line patterns using the double angle representation [55]. Their orientation estimator fits nicely into the tensor framework.

When applied to a neighbourhood of unimodal structures (lines, planes, and edges of a single orientation) the eigenvalues of the tensor allow estimation of the

---

[*] The content of this chapter has been published in: L.J. van Vliet and F.G.A Faas. Multi-orientation analysis by decomposing the structure tensor and clustering. In *ICPR18, Proceedings 18th International Conference on Pattern Recognition*, pages 856–860, Los Alamitos, Hong Kong, August 21-24 2006. IEEE Computer Society Press [146].

local dimensionality [69]. The eigenvectors yield the orientation of the corresponding image features. The fundamental property of the structure tensor that caused its enormous popularity is indisputably the fact that antipodal vectors are mapped onto the identical tensor representations, $\mathcal{M}(\boldsymbol{v}) = \mathcal{M}(-\boldsymbol{v})$. This allows averaging of mapped vectors without running into problems caused by cancelling of vectors with opposite direction.

When applying the structure tensor to multi-modal neighbourhoods, the contribution of different underlying structures, each having their own orientation, will be combined into a gradient-squared weighted averaged orientation. This average orientation is in general different from the orientation of any of the contributing structures. Hence, the real underlying structures cannot be disentangled. This is the major shortcoming of the structure tensor. To disentangle contributions based on their orientation one may employ a filter bank with a very large number of slightly rotated orientation-selective filters [51]. However, in many practical situations (corners, T-junctions, Y-forks, X-sings), a decomposition into two or three orientations and a background term is sufficient.

This paper proposes a novel method that combines unsupervised pattern recognition with image processing. It employs cluster analysis to divide the local gradient vectors that would normally construct a single tensor into a limited number of clusters. For each cluster in that neighbourhood we construct a separate tensor. Proper clustering groups vectors that resemble each other and therefore belong to the same image feature. The sum of structure tensors per cluster is identical to the standard structure tensor. In subsection 5.2 we will first introduce the mapping that constitutes the structure tensor and present a the algorithm for decomposing the structure tensor into representations for each individual oriented structure. Subsection 5.3 presents an evaluation of the decomposition method applied to crossing line structures in 2D. Subsection 5.4 discusses the extension to 3D, the parameters, and the angular resolution.

5.2 Joint spatial-feature method

Combining image processing and pattern recognition enables joint spatial and feature-space methods in solving difficult problems in image analysis such as the segmentation and analysis of complex neighbourhoods including X-sings, T-junction, and Y-forks.

5.2.1 Structure tensor

Gradient vectors in D-dimensional space can be mapped to a representation in which antipodal vectors become identical. The orientation tensor $\mathcal{T}$ is a mapping with $\mathcal{T} : \mathbb{R}^D \to \mathbb{R}^{D\times D}$

$$\mathcal{T}(\boldsymbol{g}) = \boldsymbol{g}\boldsymbol{g}^T, \quad \text{with} \quad \boldsymbol{g} = \nabla I \tag{5.1}$$

The structure tensor is a linear combination, i.e. a weighted sum, of orientation tensors in a local neighbourhood.

$$\overline{\mathcal{T}}(\boldsymbol{g}) = \overline{\boldsymbol{g}\boldsymbol{g}^T} \tag{5.2}$$

The overhead bar denotes local averaging. In 2D, the tensor becomes

$$\overline{\mathcal{T}_{2D}}(\boldsymbol{g}) \begin{pmatrix} \overline{g_1^2} & \overline{g_1 g_2} \\ \overline{g_1 g_2} & \overline{g_2^2} \end{pmatrix} \tag{5.3}$$

The directional gradient-power in direction ϕ denoted by the vector $\hat{\boldsymbol{n}} = (\cos\varphi, \sin\varphi)^T$ is

$$\begin{aligned} \mathcal{G}_\varphi &= \hat{\boldsymbol{n}}^T \,\overline{\mathcal{T}}_{2D}\, \hat{\boldsymbol{n}} \\ &= \frac{1}{2}(\overline{g_1^2} + \overline{g_2^2}) + \frac{1}{2}(\overline{g_1^2} - \overline{g_2^2}) \cos(2\varphi) + \overline{g_1 g_2} \sin 2\varphi \\ &= \frac{1}{2}(\overline{g_1^2} + \overline{g_2^2}) + \frac{1}{2}\sqrt{(\overline{g_1^2} - \overline{g_2^2})^2 + 4(\overline{g_1 g_2})^2} \cos(2(\varphi - \theta)) \end{aligned} \tag{5.4}$$

with $\varphi = \theta$ the direction that maximises $\mathcal{G}_\varphi$ and

$$\tan 2\theta = \frac{\overline{g_1^2} - \overline{g_2^2}}{2\overline{g_1 g_2}} \tag{5.5}$$

The eigenvalues λ_1 and λ_2 follow directly from Eq. 5.4 by substituting respectively $\varphi = \theta$ and $\varphi = \theta + \frac{1}{2}\pi$. The eigenvalues can be used to describe the local anisotropy A

$$A = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} \tag{5.6}$$

A closed-form solution for the eigenvalues and eigenvectors in 3D exists [51]. After that we have to rely on numerical solutions. [69] generalised the 2D estimate of local anisotropy to normalised measures for string-ness and plate-ness in 3D.

5.2.2 Decomposing the structure tensor

To avoid mixing of multiple oriented lines, edge and/or other oriented structures into a single structure tensor we need to divide the gradient vectors inside a local neighbourhood into clusters that represent a unimodal oriented line/edge structure. In the algorithm described below we choose for a clustering of the vectors after mapping. To retrieve directional information we need to inspect the symmetry of the clusters of vectors. The algorithm outlined in Algorithm 5.1 works in arbitrary dimensions. Note that the method has three parameters: the scale of the Gaussian derivative, the neighbourhood size, and the umber of clusters in k-means. The latter will be kept constant, since we apply the algorithm to the output pixels of Harris' [58] corner detector.

Algorithm 5.1

1. Compute gradient vectors at scale σ_g of a D-dimensional image I
$$\boldsymbol{g} \equiv \nabla^{(\sigma_g)} I \equiv \nabla(I * g^{(\sigma_g)})$$
with $g^{(\sigma_g)}$ an isotropic D-dimensional Gaussian kernel of scale σ_g.

2. Map the gradient vectors to an orientation tensor representation using a mapping $\mathcal{M} : \mathbb{R}^D \rightarrow \mathbb{R}^{D \times D}$
$$\mathcal{M}(\boldsymbol{g}) = \|\boldsymbol{g}\|^{-n} \boldsymbol{g}\boldsymbol{g}^T$$
with n to select the normalisation. Note that structure tensor has no built-in normalisation ($n = 0$), but the Knutsson mapping [73, 72, 110] normalises the dyadic vector product to satisfy unique stretch property, i.e. $\|\delta\mathcal{M}(\boldsymbol{g})\| = c\|\delta\boldsymbol{g}\|$ for $\|\boldsymbol{g}\| = const$. We set $n = 1$, which maps the gradient vectors of a region with constant slope (ramp edge) onto an isotropic cluster.

3. Put the unique elements of $\mathcal{M}_{ij}(\boldsymbol{g})$, i.e. the $\frac{1}{2}D(D+1)$ upper diagonal elements satisfying $i + j \leq D + 1$, in a column vector $\boldsymbol{v}$
$$\boldsymbol{v}^T = (\mathcal{M}_{1,1} \ldots \mathcal{M}_{1,D}, \ldots, \mathcal{M}_{i,1} \ldots \mathcal{M}_{i,D-i+1}, \ldots, \mathcal{M}_{D,1})$$

4. Define a local neighbourhood of tensors $\mathcal{M}(\boldsymbol{g}_i)$, with $i = 1, \ldots, N$. Store the corresponding vectors $\boldsymbol{v}_i$ in a matrix V.
$$V^T = (\boldsymbol{v}_1\, \boldsymbol{v}_2 \ldots \boldsymbol{v}_N)$$

5. Apply a k-means clustering to the collection of mapped gradient vectors. The initial value for k is three, i.e. one higher than the amount of clusters to be detected at a corner, T-junction, or X-sing. The last cluster should collect the background pixels, which manifest themselves as small random vectors. The k-means clustering is applied to matrix V, in which each row is a point in $\frac{1}{2}D(D+1)$-dimensional space
$$\boldsymbol{\omega} = kmeans(V, k)$$
with ω the vector with labels (indicating the clusters) corresponding to the rows of V.

6. The gradient vectors g_i and the orientation tensors $\mathcal{M}(\boldsymbol{g}_i)$ inside a local neighbourhood are now divided into k clusters. Compute the structure $\mathcal{M}_c(\boldsymbol{g})$ for each cluster c, where label c ranges from 1 to k.

7. Compute the system of eigenvectors and eigenvalues of $\mathcal{M}_c(\boldsymbol{g})$ for each of the k classes.

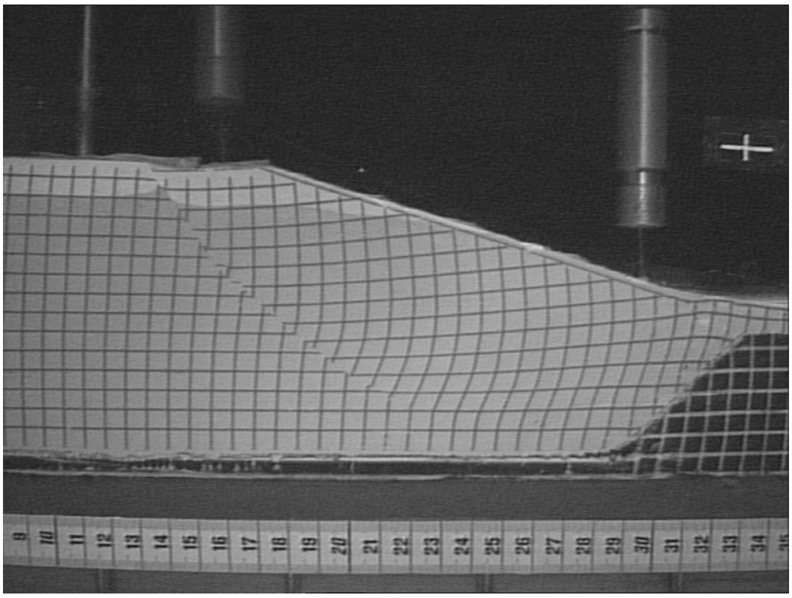


Figure 5.1: Input image of a deformed miniaturised clay-dike model with a superimposed grid. Courtesy of GeoDelft, The Netherlands.

5.3 Results

We illustrate this novel technique by applying it to the image depicted in Fig. 5.1. To assess the shear deformations of a miniaturised clay-dike model after applying a 300 fold gravitation pull, we need accurate orientation estimation of both grid axes at each crossing. The output of the procedure for $\sigma_g = 0.9$ and a neighbourhood size of 19×19 pixels is depicted in Fig. 5.2. The crossings can easily be detected using the structure tensor based corner detector by [58]. After this we apply the proposed algorithm to all crossings found. The algorithm has two important scale parameters: the scale of the gradient operator σ_g and the size of the local neighbourhood, i.e. the size of the tensor smoothing.

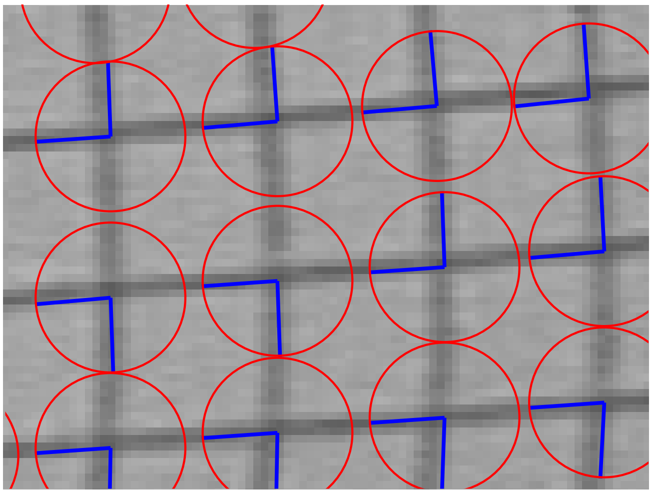
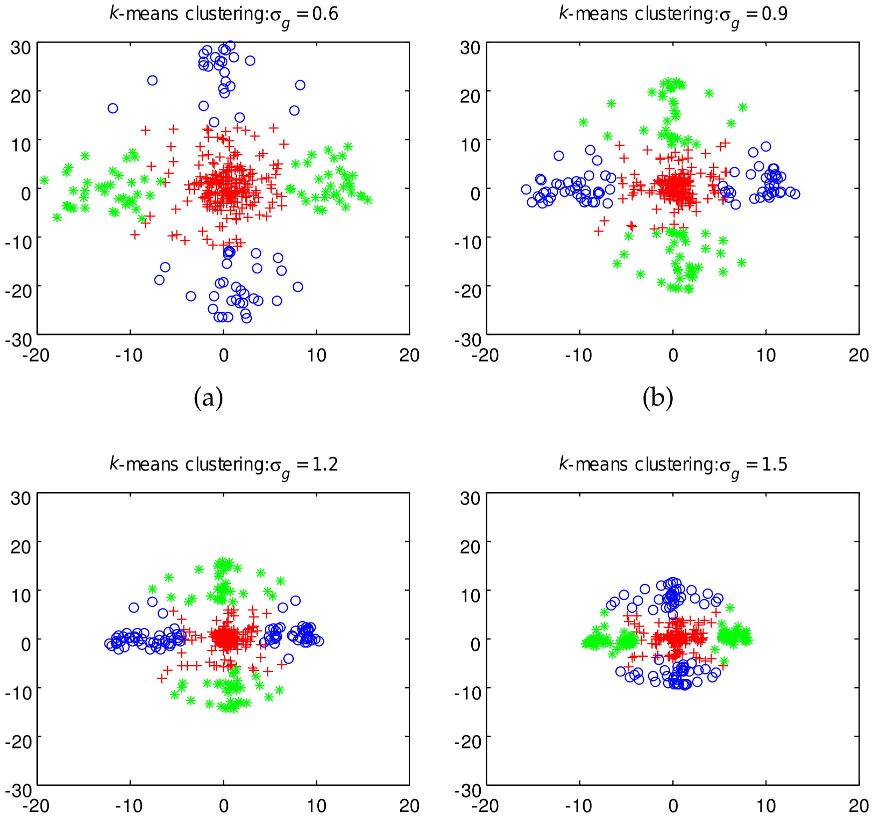


Figure 5.2: Results superimposed on the scalar input image. Parameters: $\sigma_g = 0.9$, size 19×19 pixels, $k = 3$

Table 5.1: Eigenvalues, anisotropy and orientation of the structure tensor applied to the clusters for various sizes of the Gaussian derivatives (c.f. Fig. 5.3). The neighbourhood size is 19×19.

σ_g	label	λ_1	λ_2	A	φ	σ_g	label	λ_1	λ_2	A	φ
0.6	*	11.65	1.00	0.84	0.06	1.2	*	7.99	0.30	0.93	0.07
	+	3.19	1.33	0.41	1.54		+	1.32	0.99	0.14	-1.51
	o	21.35	0.59	0.95	-1.53		o	10.79	0.55	0.90	-1.54
	{*,+,o}	5.67	3.27	0.27	-0.78		{*,+,o}	3.06	2.54	0.09	-0.70
0.9	*	9.98	0.62	0.88	0.05	1.5	*	6.51	0.27	0.92	0.06
	o	14.75	0.55	0.93	-1.53		+	1.08	0.92	0.08	1.24
	+	1.76	1.16	0.21	1.45		o	7.71	0.64	0.85	-1.54
	{*,+,o}	4.22	2.89	0.19	-0.76		{*,+,o}	2.38	2.19	0.04	-0.35



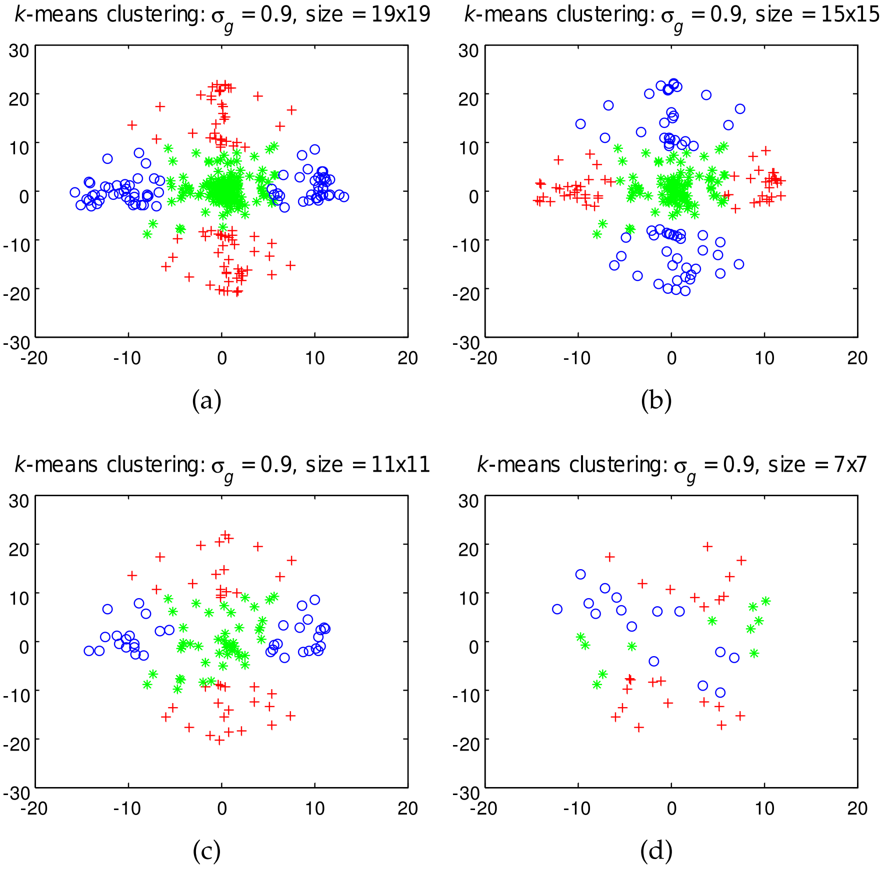
(a) (b)

(c) (d)

Figure 5.3: Scatter plot of the endpoints of the gradient vectors in a local neighbourhood after clustering (labels are assigned arbitrary). Parameters: (a) $\sigma_g = 0.6$; (b) $\sigma_g = 0.9$; (c) $\sigma_g = 1.2$; (d) $\sigma_g = 1.5$; neighbourhood size 19×19.

Table 5.2: Eigenvalues, anisotropy and orientation of the structure tensor applied to the clusters for various sizes of the local neighbourhood (c.f. Fig. 5.4). The size of the Gaussian derivatives is 0.9

size	label	λ_1	λ_2	A	φ	size	label	λ_1	λ_2	A	φ
19 × 19	+	14.50	0.54	0.93	-1.54	11 × 11	o	7.47	1.25	0.71	-0.30
	*	1.67	1.17	0.18	1.44		*	9.76	0.57	0.89	1.08
	o	9.98	0.62	0.88	0.05		+	13.43	0.99	0.86	-1.30
	{*,+,o}	4.22	2.89	0.19	-0.76		{*,+,o}	7.63	3.52	0.37	-0.74
15 × 15	+	9.80	0.74	0.86	0.04	7 × 7	+	11.72	1.43	0.78	1.44
	*	2.12	1.41	0.20	1.26		*	8.31	1.16	0.76	0.44
	o	13.99	0.67	0.91	-1.54		o	8.26	1.06	0.77	-0.89
	{*,+,o}	4.98	3.22	0.22	-0.74		{*,+,o}	7.63	3.52	0.37	-0.74



(a) (b)

(c) (d)

Figure 5.4: Scatter plot of the endpoints of the gradient vectors in a local neighborhood after clustering (labels are assigned arbitrary). Parameters: $\sigma_g = 0.9$; (a) size 19×19; (b) size 15×15; (c) size 11×11; (d) size 7×7.

5.3.1 Gradient scale dependency

To study the dependency on the scale of the gradient operator we apply the method to images of scale $\sigma_g = \{0.6, 0.9, 1.2, 1.5\}$. We keep the neighbourhood size constant around a randomly selected cross-point of Fig. 5.1. The results are depicted in Fig. 5.3. The ordering of the clusters is random, due to the random initialisation of the k-means clustering algorithm. From the systems of eigenvalues and eigenvectors after clustering one can derive that the orientation and anisotropy of the clusters remain stable (c.f Table 5.1).

5.3.2 Neighbourhood scale dependency

To study the dependency on the size of the local neighbourhood we reduce the size from 19×19 to 7×7. The scale of the gradient operator was kept constant at $\sigma_g = 0.9$. This size was selected because it corresponds critical sampling of the Gaussian kernel [147] and therefore causes no spurious blurring of high-frequency image features. The results are depicted in Fig. 5.4. The ordering of the clusters is again random, due to the random initialisation of the k-means clustering algorithm. From the systems of eigenvalues and eigenvectors after clustering one can derive that the orientation and anisotropy of the clusters remain stable (c.f Table 5.2) for tensor sizes of 19×19 down to 15×15. A further decrease of the size yields a decomposition in which the clusters have approximately the same anisotropy. A 7×7 neighbourhood clearly does not satisfy the constraint that the vectors must be independent of each other.

5.4 Discussion and conclusions

Decomposing the structure tensor allows analysis and subsequent representation of multiple oriented structures inside a local neighbourhood. The method is very fast, robust, and offers an excellent alternative to the computationally expensive filter-bank methods [52, 51].

5.4.1 Extension to 3D

Extension of this principle to three-dimensional structures is not straightforward. In 3D we must discriminate between planar (plates and edges) and line (strings) structures. Where planar structures form compact clusters after mapping of the gradient vectors, analogous to the proposed 2-D method, this does not hold for lines. Lines form compact clusters after mapping of the directional vectors which points along the orientation of minimal amplitude variation. These directional vectors cannot be uniquely determined for symmetry reasons. However, the mapping in our algorithm overcomes this problem.

5.4.2 Parameter choice

The proposed algorithm has three parameters: the scale of the Gaussian derivatives, the neighbourhood size, and the number of clusters. The Gaussian derivative should correspond to the scale of the underlying structures. The (Gaussian) scale σ_s of lines and edges can be found by wavelet decomposition [82, 29]. Using this space-variant scale removes high-frequency noise and avoid spurious blurring, hence optimising the signal-to-noise ratio of the image. Note that the minimum size for the Gaussian derivatives is $\sigma_g \geq 0.9$. The neighbourhood size must be at least a few times larger than the apparent scale σ_a , i.e. the aforementioned scale of image structures after applying the Gaussian derivatives, $\sigma_a^2 = \sigma_s^2 + \sigma_g^2$. We define the effective spatial support of a Gaussian kernel as four times its standard deviation σ_a. The tensor smoothing and hence the neighbourhood size must be at least three times the apparent scale, hence $\sigma_t \geq 5$ or a neighbourhood size larger than 15×15. The number of clusters was set fixed to three. This is allowed since we apply the method to pre-selected pixel found by Harris' corner detection. More advanced clustering strategies taking advantage of typical cluster size and shape or pair-wise clustering methods with build-in order selection can easily replace the k-means method used so far.

5.4.3 Angular resolution

The angular resolution of the proposed algorithm is defined as the minimal difference in orientation between two overlapping lines that can be discriminated by clustering. Filter-bank methods for characterisation of complex neighbourhoods, neighbourhoods that consist of multiple instances of simple structures such as lines and edges, employ rotated version of an orientation selective filter. These methods exhibit the property that the angular resolution adheres to an uncertainty principle. The product of angular resolution and neighbour size is constant [52]. The proposed method with $\sigma_g = 0.9$ and a neighbourhood size of 19×19 yields an angular resolution of approximately 25 degrees. Note that we need resolution in amplitude to separate structures from background and we need resolution in orientation to distinguish multiple instances of oriented structures.

Multi orientation analysis combining the structure tensor and unsupervised clustering.

6

abstract
The structure tensor yields an excellent characterisation of the local dimensionality and the corresponding orientation for simple neighbourhoods, i.e. neighbourhoods exhibiting a single orientation. If the analysis window contains contributions of multiple orientations, the output of the structure tensor cannot be used to find the properties of the constituents. We show that we can disentangle multi modal structures if the analysis window (tensor scale) is much larger than the gradient scale. A continuous orientation representation is used to derive a distance measure indicating the orientation pureness of each pixel. Based on this measure each point is assigned to the purest point in the local neighbourhood likely to originate from the same mode. The orientation vectors of these purest points are used to describe the local orientation structure. Clustering of these vectors in the continuous orientation representation allows identification of multiple orientations. The orientation vectors in each cluster can be separately analysed using the structure tensor approach. Proper clustering yields an estimate of the corresponding orientation.

6.1 Introduction

The structure tensor is a powerful tool for the analysis and characterisation of structures in multi-dimensional images. It was developed independently in [8] and [58] to solve two completely different tasks in respectively image processing and computer vision. Without formulating a tensor and before it was discovered, Kass and Witkin [67] presented an algorithm for estimating the average local orientation in 2D images of line patterns using the double angle representation [55]. Their orientation estimator fits nicely into the tensor framework.

When applied to a neighbourhood of unimodal structures (lines, planes, and edges of a single orientation) the eigenvalues of the tensor allow estimation of the local dimensionality. The eigenvectors yield the orientation of the corresponding image features. The fundamental property of the structure tensor that caused its enormous popularity is indisputably the fact that antipodal vectors are mapped onto the same point. This allows averaging of mapped vectors without running into problems caused by cancelling of vectors with opposite direction. The structure tensor can also be used to construct a powerful salient point detector [58] that has become a standard tool in many computer vision applications. If the structure tensor is applied to multi-modal neighbourhoods, the contribution of the various orientations cannot be disentangled. This is one of the major shortcomings of the structure tensor. To disentangle contributions based on their orientation one must employ a filter bank with a large number of slightly rotated orientation selective filters [52, 46]. However, for many practical situations decomposition into two or three orientations and a background term is sufficient. This paper presents a novel joint spatio-feature clustering method to decompose the local neighbourhood, that would normally yield a single tensor, into a limited number of clusters. For each cluster in that neighbourhood we construct a separate tensor. Proper clustering results in vector ensembles in which the contributing vectors belong to the same underlying image feature.

In section 6.2 we present the building blocks of our method for orientation measurements in multimodal regions followed by the outline of the method. In section 6.3 we demonstrate the accuracy of the method on artificial data and the performance on real data.

6.2 Method

Image structures are often represented by vector fields, such as the gradient field. Such a vector field can be mapped to a manifold in higher dimensional space. Points on such a manifold generally can be mapped back to the original space. However mixing points on the manifold, e.g. by taking a linear combination, yields a point that is not necessarily part of the manifold. In this paper this happens when points originating from different image structures are averaged. The distance to the manifold can be used to construct a "pureness" measure. In a novel spatio-feature clustering step all points are assigned to the purest example in which spatial connectivity restricts the trajectories in feature space. After that the assumed unimodal regions are clustered to merge fragmented regions.

6.2.1 Structure Tensor

A vector field in n-dimensional space can be mapped to a representation in which antipodal vectors become indistinguishable. The orientation tensor $\mathcal{T} : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ is such a mapping:

$$\mathcal{T}(\boldsymbol{v}) = \boldsymbol{v}\boldsymbol{v}^T \tag{6.1}$$

Avaraging the orientation tensors in a local neighbourhood leads to the structure tensor:

$$\overline{\mathcal{T}}(\boldsymbol{v}) = \overline{\boldsymbol{v}\boldsymbol{v}^T} \tag{6.2}$$

where the overhead bar denotes local averaging, e.g. by Gaussian weighting. The directional power along the axis described by the unit vector $\hat{n}$ is given by:

$$T_{\boldsymbol{n}} = \hat{n}^T \, \overline{\mathcal{T}} \, \hat{n}. \tag{6.3}$$

In 2D $\hat{n}$ can be put into polar coordinates, i.e. $\boldsymbol{n} = [\cos(\varphi_n), \sin(\varphi_n)]^T$, reducing Eq. 6.3 to

$$\begin{aligned} T_{\boldsymbol{n}} &= \frac{1}{2}(\overline{v_x^2} + \overline{v_y^2}) + \frac{1}{2}(\overline{v_x^2} - \overline{v_y^2}) \cos(2\varphi_n) + \overline{v_x v_y} \sin(2\varphi_n) \\ &= \frac{1}{2}(\overline{v_x^2} + \overline{v_y^2}) + \frac{1}{2}\sqrt{(\overline{v_x^2} - \overline{v_y^2})^2 + 4(\overline{v_x v_y})^2} \cos(2(\varphi_n - \theta)) \end{aligned} \tag{6.4}$$

with θ the orientation that maximises $T_{\boldsymbol{n}}$. Now θ is given by

$$\tan 2\theta = \frac{2\overline{v_x v_y}}{\overline{v_x^2} - \overline{v_y^2}} \tag{6.5}$$

The eigenvalues λ_1 and λ_2 follow directly from Eq. 6.4 by substituting respectively $\varphi_n = \theta$ and $\varphi_n = \theta + \frac{1}{2}\pi$. A closed-form solution for the eigenvalues and eigenvectors in 3D is given by Van Kempen [69]. However for higher image dimensionalities one has to rely on numerical solutions. As a whole this tensor describes the local vector field characteristics, i.e. the eigenvalues indicate the underlying certainty of the vector structure to be aligned with the respective eigenvector directions.

A common way to describe the orientation structure of unimodal structures is based on the gradient structure tensor (GST)

$$\overline{\mathcal{T}} = \overline{\mathcal{T}}\left(\nabla_{\sigma_g} I\right) \tag{6.6}$$

where $\nabla_{\sigma_g} I$ denotes the gradient vector field and σ_g the scale of the Gaussian derivative kernels.

6.2.2 Vector field

We require a vector representation which yields compact clusters around the orientation axis of the structure. Compact in the sense that the vectors are closely packed, and in which both antipodal vectors represent the orientation axis of the underlying unimodal structure. The compactness is required to keep a subsequent clustering step as simple and reliable as possible.

In 2D, the gradient vector is the obvious candidate for the vector field. The required vector field is obtained by rotating the gradient vector through an angle of $\frac{\pi}{2}$ radians to align the vector field with the orientation axis of line like structures. As such the required vector field is given by

$$\boldsymbol{v} = R_{\frac{\pi}{2}} \nabla_{\sigma_g} I \tag{6.7}$$

where $R_{\frac{\pi}{2}}$ denotes a rotation matrix for a rotation over an angle of $\frac{\pi}{2}$ rad. For planar structures the rotation of the gradient field is not necessary as it is already aligned with the orientation of the structure. Alternatives are a quadrature filter [72] and the structure tensor approach which also yield reliable results in regions where antipodal vectors are present, e.g. on opposite sides of the crest of a line. These methods are however rejected for the 2D case as they introduce additional mixing of the underlying structures.

In 3D, the gradient field is of no use as it does not yield compact clusters for line like structures. Therefore we propose to extract the vector field from the eigensystem of the structure tensor. For line like objects the direction of the vector field is given by the eigenvector associated with the smallest eigenvalue. This vector is used as the gradient energy for line like structures is confined to the subspace perpendicular to the symmetry axis of the structure. This in contrast to planar objects for which the gradient energy is centered around the axis normal to the planar structure which coincides with the symmetry axis of the structure.

Although we choose the eigenvector corresponding to the least significant eigenvalue of the structure tensor as discriptor of the local orientation, its eigenvalue is not a good measure for the certainty of the vector, i.e. the length of the vector. Therefore we set the length of the vectors to the square root of the trace of the tensor, i.e. the square root of the gradient energy present.

Mathematically the structure tensor can be written as:

$$\overline{\mathcal{T}}(\boldsymbol{v}) = R^T \Lambda R \tag{6.8}$$

with $\Lambda = diag(\lambda_1, \lambda_2 \ldots \lambda_n)$, $R = (\boldsymbol{v_1}, \boldsymbol{v_2} \ldots \boldsymbol{v_n})$ and $\boldsymbol{v_i}$ the eigenvector corresponding to eigenvalue λ_i of $\overline{\mathcal{T}}$. Assuming $\lambda_i \geq \lambda_{i+1}$, the direction of the vector field is given by the unit vector $\hat{v}_n$. Combined with the magnitude, the square root of the trace of $\overline{\mathcal{T}}$, the vector field is given by:

$$\boldsymbol{v} = \hat{v}_n \sqrt{Tr(\overline{\mathcal{T}})}. \tag{6.9}$$

Throughout this paper the use of Gaussian derivative filters is assumed. The size (standard deviation) of the Gaussian derivative and tensor smoothing kernels should be kept as small as possible to minimise the cross talk between the individual underlying structures. The standard deviation of the gradient kernels can be set to approximately one , i.e. $\sigma_g \approx 1$, [140]. Further, as a rule of thumb, the tensor smoothing should be a factor 2 to 3 larger than the gradient smoothing. This factor is on the low side compared to values reported in the literature. The value is just large enough to reliably estimate the orientation axes of the underlying structure but small enough to avoid excessive mixing of different modes.

Alternatively, for 3D line-like structures, one could derive a vector field based on the second order structure. For example, the isophote curvature method yields the vector corresponding to the axis with smallest curvature in the local isophote plane [149]. Basically, the isophote curvature method in 3D rotates the Hessian matrix to align with the gradient and the principal curvature directions in the isophote plane perpendicular to the gradient vector. This approach is however not pursued in this paper.

6.2.3 Orientation Mapping

So far we only required the clusters to be compact with respect to the orientation axis. But this is not sufficient for clustering as antipodal vectors are mapped to different points in the cluster while sharing the same orientation and possibly magnitude, i.e. this results in separate clusters. Therefore a mapping is required to remove the ambiguity between the antipodal vectors.

We define a tensor, derived from the orientation tensor, as the dyadic product normalised by the magnitude of the vector,

$$\mathcal{M}_T(\boldsymbol{v}) = \frac{\boldsymbol{v}\,\boldsymbol{v}^T}{|\boldsymbol{v}|} \tag{6.10}$$

which is a symmetric tensor, hence the subscript T, and has as such only $\frac{1}{2}n(n+1)$ independent elements, i.e. $\mathcal{M}_{ij} = \mathcal{M}_{ji}$, with n the number of dimensions. This tensor can be reduced to an orientation mapping $\mathcal{T} : \mathbb{R}^n \rightarrow \mathbb{R}^{\frac{1}{2}n(n+1)}$ satisfying the uniqueness, polar separability and uniform stretch property [73, 109]. The mapping is constructed from the unique tensor elements of Eq. 6.10

$$\begin{aligned} \mathcal{M}(\boldsymbol{v}) = |\boldsymbol{v}|^{-1}(&v_1 v_1, \ldots, v_n v_n, \\ &\sqrt{2}\, v_1 v_2, \ldots, \sqrt{2}\, v_1 v_n, \\ &\ldots \\ &\sqrt{2}\, v_i v_{i+1}, \ldots, \sqrt{2}\, v_i v_n, \\ &\ldots \\ &\sqrt{2}\, v_{n-1} v_n) \end{aligned} \tag{6.11}$$

The first n terms correspond to the n diagonal elements while the remaining terms correspond to the upper off-diagonal elements of the tensor. The factor $\sqrt{2}$ in front of the off-diagonal tensor elements makes sure the Frobenius norm of the orientation tensor is conserved in the mapping, i.e. the Euclidean norm is conserved. The Frobenius norm for a square matrix $\mathcal{A}$ is given by $\|\mathcal{A}\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} |\mathcal{A}_{ij}|^2} = \sqrt{Tr(\mathcal{A}\mathcal{A}')}$. Note that the number of independent elements can be reduced further by one more element, while preserving the aforementioned properties.

The orientation mapping of Eq. 6.11 gives a continuous but curved manifold in a higher dimensional space. For $n = 2$ the mapping is identical to the double angle representation [55]. As such all mapped vectors can be found on a cone in which the distance from the focal point of the cone is the length of the original vector. Vectors with equal magnitude are therefor mapped onto a circle embedded in a three-dimensional space. Fig. 6.1(a) shows the manifold, i.e. cone $(v_1^2, v_2^2, \sqrt{2}\, v_1 v_2)$, described by the 2D orientation mapping where the symmetry axis of the cone corresponds to the vector $(1, 1, 0)$. In this space, parallel vectors are mapped onto a straight line through the apex of the cone running over the manifold, e.g. the points on the black dashed line. Orthogonal vectors are mapped onto opposite sides of the cone, e.g. the black and gray dashed lines. In 3D the interpretation becomes more difficult and is not easily visualised although an identical formulation can be used. From geometric considerations it can be seen that averaging two points in the 3D-mapping is analog to the 2D case, see e.g. [109]. For the sake of readability we will refer to the mapped orientation vectors as s-vectors from here on.

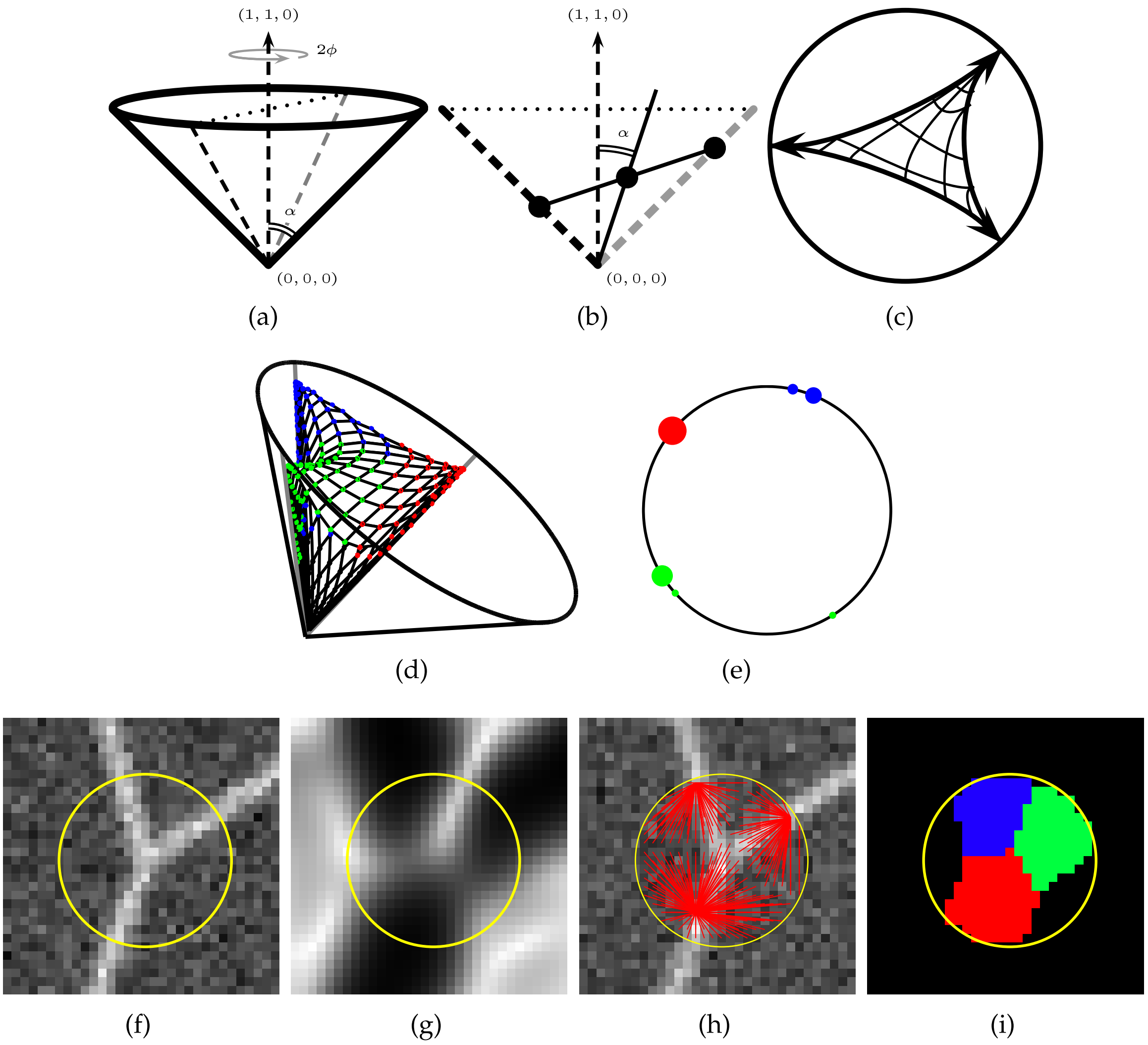


Figure 6.1: (a) Sketch of the 2D orientation mapping, the cone denotes the manifold defined by the mapping. The black and gray dashed lines denote two perpendicular orientations on the cone. The half openings angle of the cone is denoted by α and the symmetry axis off the cone is denoted by the dashed arrow. (b) Plane through the cone defined by the two perpendicular orientations in (a). The points on the dashed lines denote the mapping of two pure vectors while the third point denotes the averaged mapping. (c) Sketch of the averaged mapping around a t-junction (projection along the symmetry axis off the cone). The mesh denotes the spatial connectivity between the points. (d) Averaged vectors for a T-junction, the thick black solid lines denote the outline of the cone. The solid lines between the averaged vectors denote the spatial connectivity between the mapped points. The solid gray lines denote the pure orientations of the junction. (e) Sketch of clustered sink points on the cone mantle. (f) Artificial junction with some noise added, the yellow circle denotes the analysis window. (g) Unimodality map around the junction in (f). (h) Flow field of the points in the local analysis window. (i) Segmented analysis window.

6.2.4 Entangle and disentangle orientation vectors

A general inverse mapping from the higher dimensional space to a D-dimensional vector space is not possible as points in the higher dimensional space do not necessarily have a counter part in the original space. Further, element wise weighted sums of points on a mainfold in such a space, e.g. tensor smoothing, results in points which in general are not located on the original manifold. These two points show that local averaging of the mapping is only permitted in regions with a unimodal orientation field as only then the resulting vector is on or at least near the manifold and hence the orientation estimate can be interpreted. Obviously, the last requirement is not fulfilled for multimodal structures as averaging two s-vectors with a different orientation yields a s-vector inside the original mapping, e.g. a cone in 2D. This property is used in the next step to unmix the region where unimodal structures interact due the element wise averaging of the elements of the mapping.
Now let us define the element wise averaged mapping of Eq. 6.11 as

$$\begin{aligned}\overline{\mathcal{M}}(\boldsymbol{v}) = (&\overline{w_1 w_1}, \ldots, \overline{w_n w_n},\\ &\sqrt{2}\,\overline{w_1 w_2}, \ldots, \sqrt{2}\,\overline{w_1 w_n},\\ &\ldots\\ &\sqrt{2}\,\overline{w_i w_{i+1}}, \ldots, \sqrt{2}\,\overline{w_i w_n},\\ &\ldots\\ &\sqrt{2}\,\overline{w_{n-1} w_n}) \quad with \quad w_i = |\boldsymbol{v}|^{-\frac{1}{2}} v_i\end{aligned} \tag{6.12}$$

where the overhead bar denotes local averaging. Here a Gaussian weighting is applied with a standard deviation of σ_s.
Fig. 6.1(b) shows a cross section through the cone in the 2D orientation mapping, i.e. it depicts two perpendicular vectors mapped to opposite sides of the cone. Note that the mean of these two s-vectors due to difference in magnitude is not found on the symmetry axis of the cone but a bit off.

distance measure

As already stated, averaging the s-vectors in a multimodal region will cause the resulting s-vector to move away from the manifold induced by the mapping. We propose to use the distance from the manifold to construct a measure for the unimodality of the local environment. As the distance is proportional to the magnitude of the mapped vector we are looking into a magnitude-invariant measure. For a cone the angle between the cone mantle and a s-vector originating from the apex of the cone is equal to half the opening angle of the cone minus the angle between the s-vector with the symmetry axis of the cone. Note that the opening angle of the orientation mapping is different from mapping to mapping but constant in each mapping. Now let us define the S-vector corresponding to the symmetry s-vector of the orientation mapping as

$$S_i = \begin{cases} 1 & \text{if} \quad 0 < i \leq n, \\ 0 & \text{if} \quad n < i \leq \frac{1}{2}n(n+1). \end{cases} \tag{6.13}$$

i.e. an s-vector which is one on the elements corresponding to the diagonal elements of the tensor representation and zero for the off-diagonal elements. This s-vector has a fixed angle with all vectors on the manifold defined by the mapping. Now we introduce a distance map given by the angle between the averaged s-vectors denoted by $\overline{\mathcal{M}}$ and the symmetry axis of the cone denoted by S, i.e.

$$\begin{aligned} D(\overline{\mathcal{M}}, S) &= \alpha_n - \arccos\left(\frac{S \cdot \overline{\mathcal{M}}}{\|S\| \, \|\overline{\mathcal{M}}\|}\right) \\ &= \alpha_n - \arccos\left(\frac{1}{\sqrt{n}\,\|\overline{\mathcal{M}}\|} \sum_{i=1}^{\frac{1}{2}n(n+1)} \overline{\mathcal{M}}(\boldsymbol{v})_i\right) \end{aligned} \tag{6.14}$$

with α_n half the opening angle of the cone, where n denotes the dimensionality [109]. This distance measure depends on the angular separation of the mixed s-vectors as well as the relative magnitude of the vectors. In 2D for example, mixing two s-vectors with equal magnitude originating from two perpendicular vectors will result in distance of 45^o i.e. α_n whereas two s-vectors originating from two vectors under an angle of 45^o will result in distance of $\approx 10^o$.

Note that Eq. 6.14 can also be derived directly from an element averaging of Eq. 6.10 given by

$$\overline{\mathcal{M}}_T(\boldsymbol{v}) = \overline{\left(\frac{\boldsymbol{v}\,\boldsymbol{v}^T}{|\boldsymbol{v}|}\right)}. \tag{6.15}$$

The angle between the manifold and the element-wise averaged tensors $\overline{\mathcal{M}}$ is given by

$$D(\overline{\mathcal{M}}, S) = D(\overline{\mathcal{M}}_T, \mathcal{I}) = \alpha_n - \arccos\left(\frac{\mathcal{I} \cdot \overline{\mathcal{M}}_T}{\|\mathcal{I}\|_F \, \|\overline{\mathcal{M}}_T\|_F}\right) \tag{6.16}$$

where $\mathcal{I}$ denotes the identity matrix and $\| \cdot \|_F$ the Frobenius norm.

Fig. 6.1(b) shows the result of averaging two s-vectors that correspond to perpendicular vectors in a 2D image sub-space. α denotes the angle between the symmetry axis of the cone and the averaged s-vector, i.e. the distance is equal to $\alpha_n - \alpha$.

The sketch in Fig. 6.1(c) depicts a top view of locally averaged s-vectors to a 2D T-junction. In this case three modes are indicated with the three arrow heads. The spatial connectivity between the mixed s-vectors is depicted by the mesh. Due to averaging all s-points, i.e. mesh points can be found in the cone or on the cone wall in case of a pure unimodal region. Note that the mesh due to averaging between three modes can be found in the convex hull described by the three pure orientations at least in a noiseless case. More general, the result of the weighted mean of a set of initially pure s-vectors is confined to the convexhull described by the vectors from which this weighted mean is constructed. Fig. 6.1(d) presents a 3D view of the mesh generated in the local neighbourhood indicated by the yellow circle centered on the fork in the image depicted in Fig. 6.1(f). The black dots indicate the low magnitude vectors close to the apex of the cone while the coloured dots indicate the stronger vectors. From this example we can clearly see the tendency of the s-vectors to form laminae through the cone connecting the pure orientations. Further we can see that the space inside the laminae is almost empty as the probability that a point belonging equally to three modes is very unlikely, i.e. it is much more likely that a point

belongs more or less to two pure structures than to three. This can of course be generalised, i.e. the probability to have equal contributions from $m+1$ pure orientations is much lower than the chance to have equal contributions form m pure orientations. This of course in the neighbourhood of a complex multimodal structure.
Fig. 6.1(g) shows the distance map of the image in Fig. 6.1(f). It clearly shows that the distance values are maximal at the junction and drops towards the unimodal regions.

region selection

The downhill gradient of the local distance landscape points to regions with less interaction between modes, i.e. a smaller distance to the manifold. As such we can apply a gradient descent type of algorithm, i.e. we connect each point in the local neighbourhood to its downhill minimum [152]. The downhill minimum is found by stepping to the neighbour with the lowest "distance" value and continuing so until all neighbours have a distance greater or equal to the present point. Now clusters are formed by points that share the same downhill minimum, i.e. downhill minima work as sinks. Ideally, each region corresponds to a single unique mode. However this is often not the case if noise and other imperfections such as curved linear structures can cause fragmentation of modes. Therefore we have to include a step to merge fragmented structures (or structures with equal orientation). Fig. 6.1(h) shows the flow field of the image in Fig. 6.1(f). This figure shows that all points in the local neighbourhood flow to one of the three sink points. Note that the sinks on the border of the local neighbourhood are not necessarily preserved if the neighbourhood size is increased. Also note that the number of sinks gives a upper limit for the number of linear structures present in the local neighbourhood. Similarly, one can never find more orientations than the number of clusters in the local neighbourhood.

6.2.5 clustering

To cluster the fragmented regions we use a weighted k-means algorithm. The c data points are represented by the orientation vectors of the sinks and the weights are set to the size of the corresponding regions. As the algorithm is not guaranteed to return a global optimum we obtain $m+1$ solutions for $m+1$ different initialisations, in here we set m to the minimum of 10 and $c!$ (in the latter case all permutations are evaluated). The first initialisation of the algorithm is done with the sinks of the k largest regions. The subsequent m initialisations are done by randomly picking k sinks from all sinks. The best clustering, i.e. most compact, is accepted.

6.2.6 measurement

After segmentation and labelling we obtain k unimodal regions. Now we can apply the structure tensor to the regions separately. This gives an unbiased orientation estimate as long as the orientation vectors are evenly distributed around the true orientation. In Fig. 6.1(e) a sketch is shown of the orientation vectors describing the sinks, i.e. in the figure all vectors are mapped to the same magnitude ending up on a circle. The size of the dots indicates the size of the region. The figure indicates

6 sinks forming three solid compact clusters. One of these sinks is a small outlier cluster which is assigned to the green cluster. Note that the assignment of such a cluster is not significant to the end results as its weight in constructing the structure tensor is low compared to the rest of the cluster.

6.2.7 Algorithm

To summarise, we present an outline of the algorithm including the parameters involved at each step along the way.

1. First a vector field is created which is aligned with the orientation of the structures of interest. In 2D this field is derived directly from the gradient field (Eq. 6.7). Hence only a single scale σ_g is needed for computing the gradient. In higher dimensions the vector field is obtained from the eigensystem of the structure tensor (Eq. 6.9), which involves setting a gradient scale, σ_g, and tensor scale, σ_t. To limit the cross-talk between individual modes small, the smoothing parameters should be kept as small as possible. For moderate noise conditions we propose the use of a $\sigma_g \approx 1$ and $\sigma_t \approx 2\sigma_g$.

2. From this representation a continuous orientation representation is derived which is further smoothed in the local environment to obtain information on the pureness of the underlying structure, see Eq. 6.12. This introduces an additional smoothing scale, $\sigma_s \approx 3\sigma_g$, note that this smoothing is additional to the already performed smoothing, i.e. the cumulative smoothing is approximately $\sqrt{(\sigma_g^2 + \sigma_t^2 + \sigma_s^2)} \approx 4\,\sigma_g$.

3. From this averaged orientation mapping a distance map is constructed describing the pureness of the image point in question, see Eq. 6.14.

4. In a window with radius r, we apply an iso-data threshold [107] on the square-root of the gradient energy to separate orientation carrying pixels from background pixels. This step basically simplifies the clustering as it removes the low magnitude vectors. In 2D this removes points around the apex of the cone. This step can be applied globally to the image whenever properties like shading and image blur are constant over the image.

5. The distance map, Eq. 6.11, is used to find regions around the local minima in the distance map, i.e. regions contain connected points sharing the same downhill minimum in the distance map. The properties of the minima are used to describe the respective regions, thereby significantly reducing the number of vectors to be processed.

6. To combine regions with the same orientation the descriptors of the regions are clustered in k clusters by means of a weighted k-means clustering algorithm. The weight of each region is equal to the number of points. Such regions are represented by its sink (see step 5). This step can be replaced by a more complex clustering algorithm which for example could pick the number of modes automatically.

7. For each of the k clustered regions a structure tensor is constructed based on the vector field calculated in step one, in which the tensor averaging is uniform over the cluster (except for the pixels below the iso-data threshold which have zero weight). For each of these tensors the eigenvector corresponding to the dominant eigenvalue is calculated and used as orientation descriptor of the region.

6.3 Results

In this section we first apply our method to artificial bifurcations to determine the accuracy and precision of method. Secondly the method is applied to some real-world images to show possible applications.

6.3.1 Y-fork

To quantify the performance of the method it is applied to artificial Y-fork images, i.e. a bifurcation in which the angle between the "standing leg", A, and the other two "arms", B and C, is identical. As such the geometry of this structure can be described by a single angle ϕ, i.e. $\phi = \angle AB = \angle AC$ while $\angle BC = 2(\pi - \phi)$. In $3D$ the entire structure is located in a single randomly oriented plane.
The Y-forks where generated for angles $\phi \in [0, \pi]$ with a random pose, i.e. with random subpixel position and random orientation for the structure as a whole. The lines, "arms", have a Gaussian line profile, i.e. $I(s) = \exp(-\frac{s^2}{2\sigma_l^2})$ with s the distance from the centre of the line. We added respectively {10,15,20} dB Gaussian noise to the images. Here the SNR is defined as $20log(\sigma_n^{-1})$ dB with σ_n the standard deviation of the noise and the implicit assumption of a line contrast equal to one, i.e. the difference between foreground and background intensity is assumed one. For the radius of the analysis window we used the values of 5, 10 and 15 pixels respectively. The measurements are performed 200 times for each combination, e.g. noise level, angle ϕ and radius of the analysis window. Throughout the experiment the values for $\{\sigma_l, \sigma_g, \sigma_t, \sigma_s\}$ were fixed at $\{1, 1, 2, 3\}$
The results are shown in Fig. 6.2, the left and right column respectively show the results for the $2D$ and $3D$ case. The figures on the first row, Fig. 6.2(a-b), show the mean angular deviation, $\overline{d}$, for the three arms where d is given by

$$d = \underset{P_{ij}}{min} \sum_{i=1}^{3} \arccos |u^T(\phi_i) u(\theta_{j(i)})| \tag{6.17}$$

with ϕ_i the true and θ_j the measured orientations and $u(\cdot) = [cos(\cdot), sin(\cdot)]^T$. The minimisation is over all permutions, P, of the measured orientations with respect to the true orientations.
The second row of figures, Fig. 6.2(c-d), shows the bias in the estimated orientation for each arm, i.e. A, B and C in the case of 15 dB noise. This bias measure is denoted by $\overline{d_i}$ where d_i is given by

$$d_i = sign(u(\phi_i) \times u(\theta_{j(i)})) \arccos (|u^T(\phi_i) u(\theta_{j(i)})|) \tag{6.18}$$

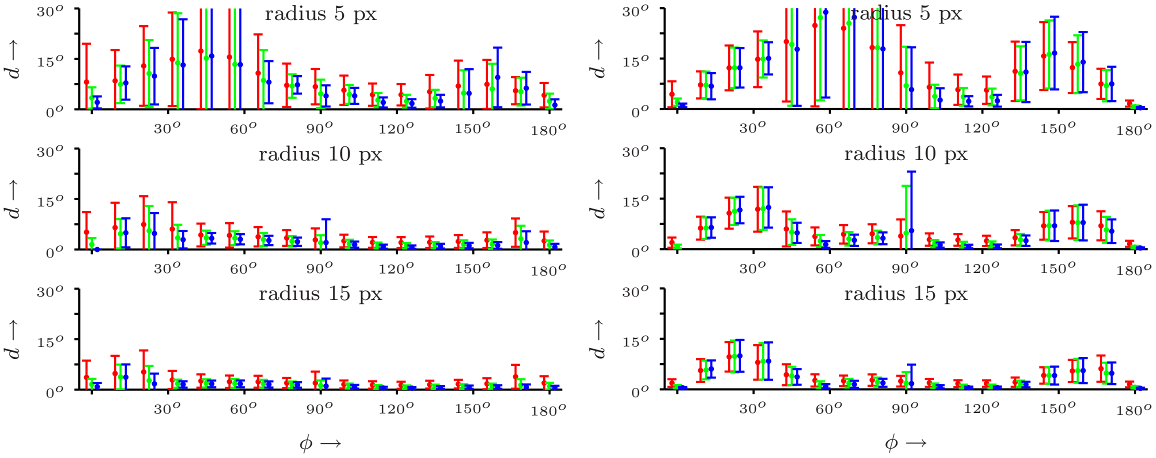
here the first factor determines if the angular deviation is clockwise or counter clockwise and the second factor determines the angular deviation itself. The permutation between true and measured orientation is chosen in the same way as in the previous experiment. The $\{$red,green,blue$\}$ lines respectively correspond to the arms $\{A, B, C\}$. Note that the colour code is different between the figures in the first and second row, i.e. in the first row the colours denote noise levels while in the second row the colours denote the individual arms in a specific noise situation.

The third row, Fig. 6.2(e-f), shows cumulative histograms of the number regions before and after clustering in the local neighbourhood. The number of regions before clustering is shown in the upper figure while the number of regions after clustering is shown in the lower figure of each pair. The conditions are the same as for the bias experiment, i.e. a noise level of 15 dB and a radius of 10 pixels for the local neighbourhood. The post clustering figures are basically collapsed versions of their pre clustering counter parts, i.e. with less starting regions than the required number of orientations the algorithm will return the initial regions and in all other cases exactly the required number.

For the interpretation of the results it is important to realise that leg A coincides with the symmetry axis of the structure. Furthermore, there are three degenerate structures, i.e. structures in which the orientation of two or more lines coincide. These structures are described by $\phi \in \{0, \frac{\pi}{2}, \pi\}$. Another point to keep in mind is the fact that the results in the upper two rows of figures only refer to cases which yielded the required number of orientations, i.e. whenever the algorithm yielded less orientations than required the results where ignored in the analysis.

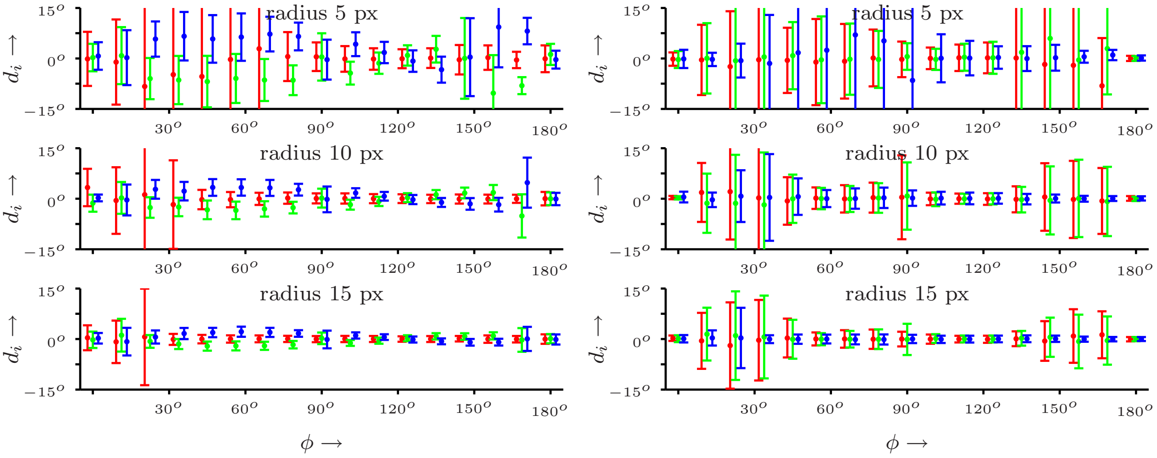
The most striking in the results is probably the bias behaviour in the 2D case. Normally the GST is unbiased however in this case the distribution of the orientations in the assigned regions can become truncated. This is a result of the way k-means assigns clusters, which is basically a nearest mean assignment. This hard assignments makes a choice for a single cluster while in overlapping regions points should be assigned on the basis of the overlapping distributions of the underlying structures. Whenever this truncation is not symmetrical around the true mean it will introduce a bias. As the angular distance from arm A to arm B and C is equal, the truncation works identical on both sides of the underlying orientation distribution. Hence the results for arm A (red) should be unbiased. However this is not the case for the arms B and C (blue and green) for which the angular distance to their neighbours is in general different for each neighbour and as such also the truncation will not be symmetrical around the true mean. The absence of bias in the 3D case can be explained by the fact that in $3D$ the truncation will be highly confined to the plane of the structure, i.e. out of plane the truncation effect will drop drastically. As such the majority of the underlying distribution will be unaffected by the hard assignments of the clustering method.

The cumulative histograms show clearly the effect of the degeneracy of the structure for some cases, i.e. for $\phi = 0$ the structure resembles a single half line with a single orientation and as such often only one or two starting clusters are found. The same is true for $\phi = \pi$. For a T-junction , $\phi = \frac{\pi}{2}$, the effect is negligible as the presence of arm A probably forces a watershed between arms A and B in the distance measure.



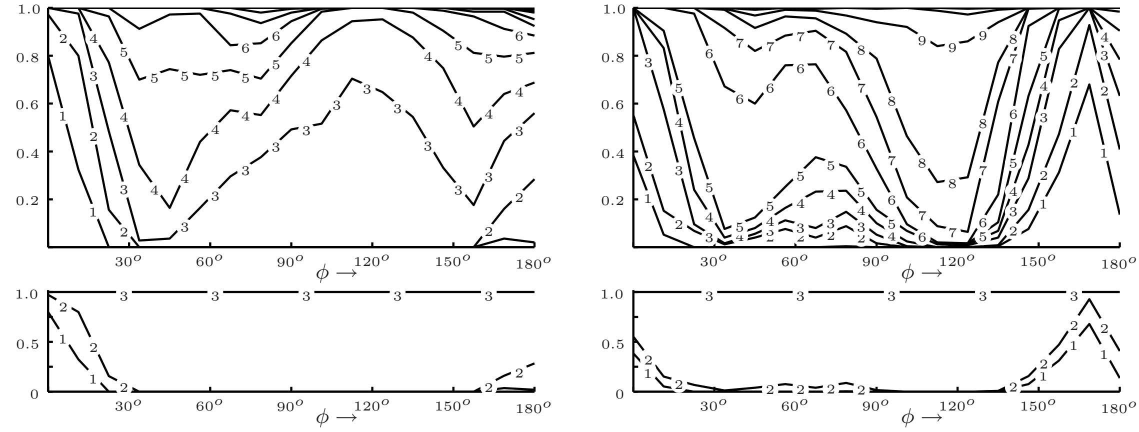
(a) 2D: angular deviation

(b) 3D: angular deviation



(c) 2D: bias

(d) 3D: bias



(e) 2D: cumulative histograms

(f) 3D: cumulative histograms

Figure 6.2: Results of the Y-fork experiment with respectively the 2D and 3D results in the left and right column. Figs. (a-b) show the mean deviation from the true orientation, the {red,green,blue} lines correspond to the {10,15,20}dB cases. Figs. (c-d) show the bias for the separate arms of the Y-fork for 15dB noise, the {red,green,blue} lines correspond to the arms $\{A, B, C\}$. Figs. (e-f) show the cumulative histograms of the number of regions before and after clustering in respectively the upper and lower cumulative histogram, for 15dB noise and r = 10 px.

6.3.2 Test images

claydike

The image in Fig. 6.3(a) shows a deformed miniaturised claydike model with a superimposed grid. In this application the accuracy and precision are important to characterise the stress and strain in the deformed grid. The measured orientations are superimposed on the image where the length of the lines indicated the size of the analysis window. For this test we put k at 2, i.e. crossings have two main orientations. The values of the parameters, i.e. $[\sigma_g, \sigma_s, r, k]$ are set respectively at {2,4,15,2}. We had to put r and σ_g to relatively high values due to scan artifacts in the image, i.e. in the image horizontal smearing is present which affects the vector field. For this test the points of interest were selected automatically [34]. In the case of simple crossings the orientations of the underlying structures are nicely detected. In the cases where the implicit assumption of a lineair structures is violated the estimates can be a bit off as well as interfering structures are present.

leaf

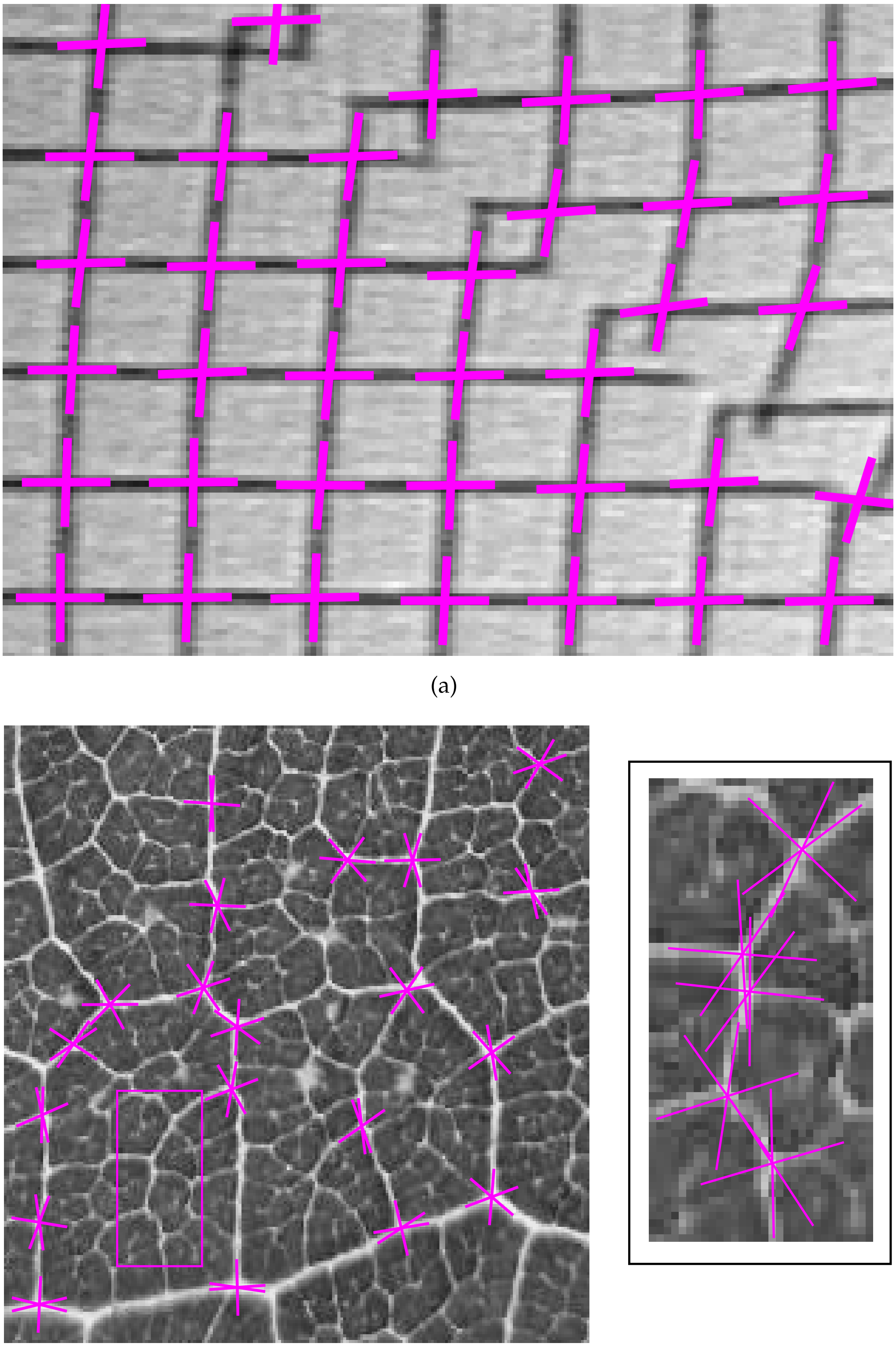
The image in Fig. 6.3(b) shows a close up of the veins in a leaf. The detected orientations are shown in the overlay as in the previous example. The rectangle indicates the location of the enlarged region on the right. The values of the parameters, i.e. $[\sigma_g, \sigma_s, r, k]$, are set respectively at {1,3,10,3}. The orientations at the selected junctions are detected nicely Although in some cases a human observer would indicate less orientations, e.g. in the case of T-junctions two orientations would be more appropriate.

checker board

The image in Fig. 6.3(c) shows a part of a checker board image. The checker board pattern gives rise to crossings, only the upper right corner has a fork structure, hence we put $k = 2$. The parameter values, i.e. $[\sigma_g, \sigma_s, r, k]$, are set at respectively {1,2,10,2}. In the insets we show a blowup of the corner for respectively $k = 2$ and $k = 3$. As can be seen in the $k = 3$ case the orientations are detected correctly. The reduction of k to two causes two regions to merge and hence the orientation measure in this new region is in some sense the weighted average of orientations in the merged regions. Note that the orientation estimate of this merged cluster is not exactly the half way orientation but is closer to the orientation of the strongest edge.

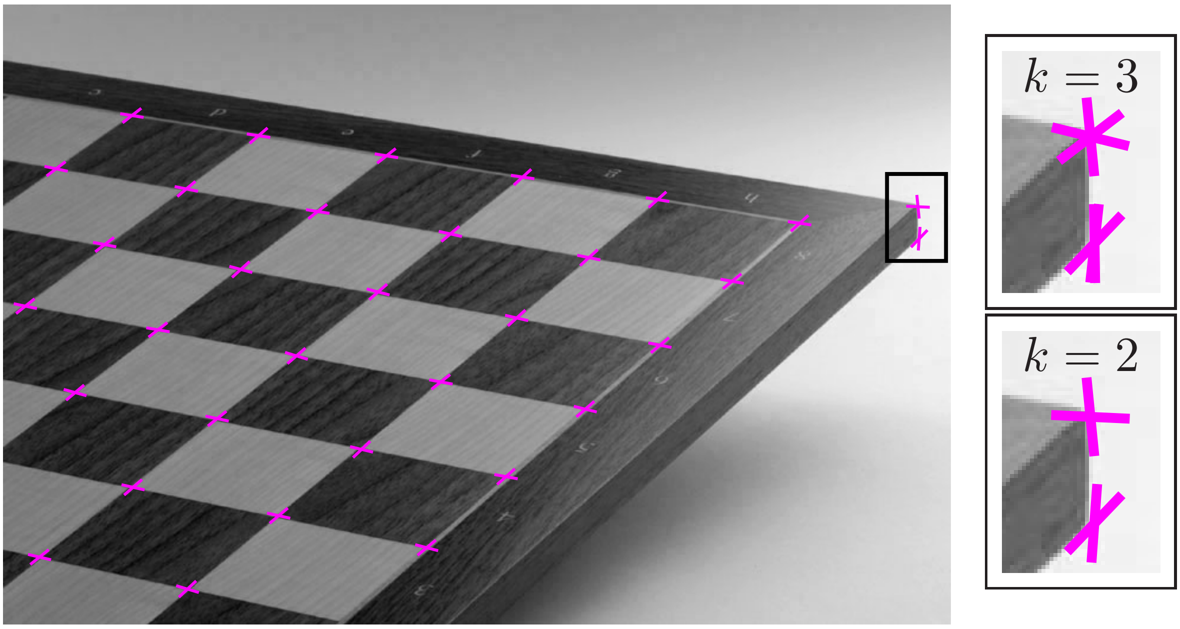
6.4 Discussion and conclusions

The traditional structure tensor approach is only valid in unimodal regions. The tensor regularisation step will mix the contributions of the underlying structures which can not (easily) be disentangled afterwards. To address this problem we introduced a distance measure which indicates the pureness of a local neighbourhood. This measure makes use of the fact that when two modes are averaged in the orientation mapping the average moves away from the manifold enforced by the mapping. Realizing that an inverse mapping is only possible for points on the manifold we



(a)

(b)



(c)

Figure 6.3: Test images with in overlay the detected orientations at a number of selected points. (a) Image of a deformed miniaturised claydike model with a superimposed grid. Courtesy of GeoDelft, The Netherlands. The parameters $\{\sigma_g, \sigma_s\, r, k\}$ set to $\{2, 2, 4, 15\}$. (b). Image showing the venation of a leaf, with $\{\sigma_g, \sigma_s, r, k\}$ set to $\{1, 3, 10, 3\}$ (c). Checkerboard image with parameters $\{\sigma_g, \sigma_s, r, k\}$ set to $\{1, 2, 10, 2\}$

assigned each pixel in the distance map, by means of a steepest descent, to the pixel in the local neighbourhood with the smallest distance to the manifold. The pixels with the same downhill minimum form in principle unimodal regions. Fragmented regions are clustered, resulting in k regions where k should be the number of orientations displayed by the underlying structures. After clustering, the structure tensor can be applied to the resulting unimodal regions (clusters).

If the number of clusters cannot be determined one could skip the clustering step which would result in a number of tensors equal to the number of regions in the local neighbourhood. This option would result in a structure like that of the filterbank approach [52, 46] but than sparse, i.e. each tensor and the size of the corresponding region could be seen as a point along the orientation axis in a filterbank. Compared to this well known method we can achieve a high angular selectivity with relatively small analysis windows. An alternative approach is that of the non-linear structure tensor [16]. Although in this method the diffusion, i.e. smoothing, is damped at discontinuities it is not able to return a multi valued description of the orientation field in the local neighbourhood.

The computational cost of the non-linear structure tensor as well as the filterbank approach is assumed to be as least as high as for our method, i.e. the filterbank needs a large amount of filters for a good angular selectivity while the non-linear structure tensor needs a large number of iterations. Our method has low memory requirements due to the sparse nature. Our method is reasonably fast as the distance map can be calculated globally. The first local step, i.e. the segmentation, is based on a steepest decent and is fast as only for a small fraction of points the entire descent has to be done. The other points can use the results from completed runs whenever they join the same path. The second local step, i.e. the clustering, is also reasonably fast as the number of regions is very small which speeds up the clustering process considerably when compared to a clustering directly on all points in the neighbourhood.

The method requires a few parameters. First of all a gradient scale and in 3D an additional tensor scale which should be kept as small as possible to keep cross talk to a minimum. Second, the smoothing in the orientation mapping should be of a much larger scale as it should "see" the different modes. Third the clustering neighbourhood, i.e. analysis window, should be larger than the spatial extend of the multimodal structure. This as the vectors on the junctions do not represent the underlying structures very well in general. Finally k sets the number of required orientations.

The bias behaviour we found in the 2D case results from truncation of the distribution of the orientation field in the analysis window. This bias could possibly be reduced by a clustering method which assigns soft labels, i.e. a method which returns the probability of a point to belong to a certain cluster.

We did not address the problem of automatically picking the number of clusters k, although an upper limit is set to the dimensionality by the number of regions in the local neighbourhood. Furthermore, the method can easily return the results for e.g. $k = 1$ to $k = 5$ without increasing the computational cost to much. By analysing these results it could be possible to automatically pick the best value for k.

Our examples show the power of the method but we would like to stress that the method is not confined to these applications. Other possible applications include non-linear diffusion schemes, corner detection and texture segmentation and classification.

Distance transforms and path extraction

7

7.1 Introduction

Finding the fastest path through a medium from start to finish is a problem that arises in many areas of natural and engineering science. If the search space can be represented by a (bi-)directional graph a solution can be found in a rather straightforward fashion using one of the many optimisation algorithms such as A*. It becomes complicated if the space is continuous and in the presence of a space-variant speed function. In geometrical optics this problem corresponds to finding the path light follows while travelling through a medium with a space-variant refractive index. In acoustical wave propagation it produces the path that a sound wave traverses between source and receiver through a material with a space-variant acoustical impedance. Similar problems arise in other fields of science and engineering, e.g. path planning and image processing. In all cases it reduces to finding the fastest path from source to destination through a medium. Both the source and the destination are not necessarily limited to single points, but can be objects of a certain dimensionality and spatial extent, e.g. lines, planes, volumetric objects. This problem can be written in the form of the Eikonal equation, a nonlinear partial differential equation which can be derived from the Hamilton-Jacobi equations.

Solving the Eikonal equation is solving an optimisation problem, i.e. finding the fastest path between source and destination. This minimisation problem can be thought of as an outward propagating interface, a wave with arrival time $U(\boldsymbol{x})$ and propagation speed $V(\boldsymbol{x})$. For a wavefront to move with speed V an infinitesimal step dx requires an infinitesimal time-step dU or

$$dx = V dU. \tag{7.1}$$

As information is only transferred along a time line, V is assumed to be strictly positive, which leads to the Eikonal equation:

$$\begin{cases} I = |\nabla U| & \text{for } x \in \bar{\Omega} \\ U = 0 & \text{for } x \in \delta\Omega \end{cases}$$

where $I(\boldsymbol{x})$ is the inverse speed function, V^{-1}, also referred to as cost function. The cost function in geometrical optics is the refractive index, in acoustical wave propagation the impedance. Further as boundary conditions the initial front, $\delta\Omega$, is set to have a zero arrival time.

So far we assumed the speed function to be isotropic, i.e. the local speed is independent of the local propagation direction. In a more general way one can easily envision speed functions that do depend on the orientation and curvature of the path connected to the current point on the wavefront and hence orientation and curvature dependent cost functions. Such an anisotropic cost function depends on properties derived from the local neighbourhood and as such are not isotropic because, in general the content of the neighbourhood itself is not isotropic.

As the solution of the Eikonal equation is the arrival time of the wavefront, a second step is needed to obtain the minimum cost path connecting the point of interest to the initial front. It can be shown that the minimum cost path is obtained by a gradient descent from the point of interest to the initial front over the arrival time map. Like a ball, with zero inertia, rolling down hill. The zero inertia is important as otherwise the ball will try to keep moving in a straight line instead of instantaneously following the projected force of gravity.

Background

Taking the Eikonal equation as a starting point, the equation was first solved for a medium with a space-invariant cost function. For unit costs this boils down to the Euclidean distance connecting a point of interest to the initial front (i.e. the source). Hence, the arrival time and the distance travelled are identical. Similarly in optics a space-invariant refractive index causes light rays to travel along straight lines, i.e. the fastest path equals the shortest path.

Rosenfeld and Pfaltz [116] introduced the distance transform in which a local distance metric is propagated to obtain an approximate global Euclidean distance map. The only consistent and therefor valid distance functions, are the ones that form a metric. Any metric satisfies the following criteria for a distance function, $d : X \times X \to \mathbb{R}_0^+$:

$$\begin{aligned}
&d(\boldsymbol{x},\boldsymbol{y}) = 0 \text{ if and only if } \boldsymbol{x} = \boldsymbol{y} && \text{(identity of indiscernibles)}\\
&d(\boldsymbol{x},\boldsymbol{y}) \geq 0 && \text{(non-negativity)}\\
&d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x}) && \text{(symmetry relation)}\\
&d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{p}) + d(\boldsymbol{p},\boldsymbol{y}) && \text{(triangle inequality)}
\end{aligned}$$

Rosenfeld only gave exact solutions for the city block and chessboard metric which respectively correspond to the L_1 and L_∞ norm. Rosenfeld and Pfaltz [116] also introduced the octagonal distance transform which alternates between the city block and chessboard metric to partially compensate the overestimate of the former by the underestimate of the latter. To get a better approximation to the Euclidean distance or L_2 norm Montanari [91] introduced a metric, which gives the exact Euclidean distance on the grid axes and the diagonals but makes errors in between. To minimise the maximum error Borgefors [11] introduced the Chamfer metric in which the metric coefficients are chosen such that the maximum error is minimised for all directions. Danielsson [24] introduced a quasi Euclidean distance transform based on a two component descriptor for the number of horizontal and vertical steps, quasi as in a small number of cases small errors can be detected. The result is a vector at every position, hence the name vector distance transform (VDT). This method was later extended by Mullikin [94].

Parallell to the Euclidean distance transforms (EDT) the family of grey value weighted distance transforms where developed. In which not the distance is measured but the arrival time of the wavefront.

In Rutovitz [117] an algorithm was proposed in which the cumulative cost is lower along a path with low pixel values. Levi and Montanari [78] introduced the Grey-weighted skeleton for which the grey value distance is defined as the minimum cumulative grey value sum along all possible paths. Another grey weighted distance measure is that of the fall-distance, i.e. in a region which is strictly decreasing the fall-distance is given by the length of the path of decent [118, 152]. Later Toivanen [133] presented a distance transform based on grey value differences.

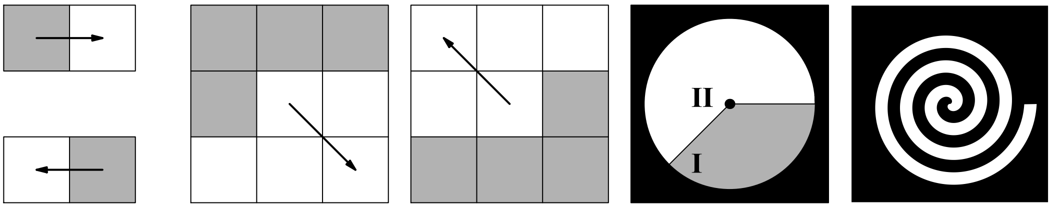
Verbeek and Verwer [139] introduced the grey value weighted distance transform (GDT) based on the chamfer distances weighted with the local grey value, an exact solution for a maximum finite difference gradient. Since the propagation directions in the grey weighted distance transform are governed by the adopted chamfer only a discrete set of propagation directions is allowed.

A method not suffering from the discrete set of propagation vectors is the fast marching method (FMP). This method locally models a planar wave onto the underlying cost function [136, 126]. Like the GDT and EDT it is not isotropic although the maximum error is an order of magnitude lower. As mentioned the Eikonal equation propagates information only forward in time, i.e. along the gradient direction of the arrival time of the wave front. This principle is used in the Euclidean distance transform as presented by Verwer et al. [142] , the GDT as well as the Fast Marching methods. The wave front is expanded by the Huygens principle. Point(s) on the leading wave front work as a source for a new wave which updates all points ahead of the wavefront. Where the principle is implemented by a Dijkstra like method [30].

To extract the minimum cost path from a given arrival time map a gradient descent has to be applied. The descent is from some point of interest on the arrival time map to some point on the initial front. The gradient can be obtained by various methods. The simplest method is the binary descent in which the descent algorithm steps to the lowest valued 8 connected neighbour until the point it self is the (local) minimum. Applied to the arrivaltime map you end up at some point on the initial front. Other methods use more regularization like the Runge-Kutta based methods.

Method

The distance transforms can be divided in chamfer, vector and fast marching based transforms. The key differences between the aforementioned distance transforms is the approximation in modelling the wave. In the chamfer methods a wave is only allowed to travel over a discrete set of vectors which start and end at grid points. The FMP method models a plane wave which also supplies the propagation direction, i.e. the propagation direction is not limited to a small set of grid-based directions.



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

Figure 7.1: (a) Propagation masks for 1D, (b) forward 2D propagation mask, (c) backward 2D propagation mask, (d) region processing for single seed, where region I and II obtain their final values in respectively the first and second pass. (e) spiral image.

The former method is inherently discrete. The latter approximates the continuous wavefront, which is a good approximation as long as the wavefront is sufficiently straight. For (strongly) curved fronts this approximation breaks down. Therefor we propose a method in which the wave is locally modelled as a spherical wave.

Another point of break down is that locally all paths are assumed straight which again is the right assumption for uniform cost functions, but for space-variant cost functions this assumption is violated and as such the minimum cost paths can be strongly curved. However this error source is not investigated further in here as the errors introduced by the modelling of the wave front dominated the total error. As a solution one could think of a method which locally solves the Eikonal equation on a planar cost function allowing the paths to bend locally. This method then converges to the classical FM method in the limit of a infinite radius of curvature of the wave front just as the spherical wave approximation.

7.2 Euclidean distance transform

For image processing tasks such as skeletonisation and segmentation one often needs the distance between objects. Therefore a big effort has been put into finding fast and accurate solutions to this problem. Finding the distance between objects is basically a minimisation problem as one tries to find the minimum distance to some initial front or source.

For a region of interest where the boundary is the initial front, a number of sequential algorithms have been developed, such as the two pass algorithm in Rosenfeld and Kak [115] and a four pass algorithm in Danielsson [24].

The two pass algorithm first makes a forward pass and then a backward pass, e.g. for 1D first from left to right and then in opposite direction. For 2D the forward sweep is from the top left to the bottom right with a causal neighbourhood, i.e. all top neighbours and the left nearest neighbour correspond to positions that have been visited in the current forward sweep. For each point on the sweep front the minimum cumulative sum is calculated for all points in the mask, i.e. the arrival times of the mask pixels are increased by the respective metric distances and the minimum value is accepted as arrival time for the pixel on which the mask is centered. The forward and backward masks in 1D and 2D are shown in Fig. 7.1(a-c) and a simple 1D example is given in in Table 7.1. This idea is easily extended to higher dimensions see e.g. Borgefors [11].

Table 7.1: Euclidean distance transform on a 1D image (1^{th} row. The second row contains the assigned distance values after the first pass and the third row the final values. See Fig. 7.1(a) for the propagation masks.

binary image	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1
1^{th} pass	*	*	*	0	1	2	3	0	1	2	3	4	5	0	1
2^{nd} pass	3	2	1	0	1	2	1	0	1	2	3	2	1	0	1

A big restriction on these sequential algorithms is the limited region of influence of the points on the sweep front. This region is bassically a cone with a 135 degree openings angle. Lets have a look at a simple example in which the boundary of the region of interest is not the initial front. Assume a circular region of interest with a single source point. The resulting distance map should be a cone with an 90 degree opening angle at the bottom centered on the seed point,see Fig. 7.1(d). The first pass will assign the correct values to area I while the second pass will do so for the union of area I dilated by the reflected set. For constrained distance transforms (also called geodesic distance transforms), many more passes through the image are needed dependent on the geometry of the accessible space. A spiral clearly shows that the required number of passes till convergence can be quite high, Fig. 7.1(e).

As the sequential algorithm is not suited very well for such a general case an ordered propagation algorithm was proposed by Verwer et al. [142]. The shortest path is found by applying a uniform cost algorithm, a special case of the A^* algorithm combined with the bucket sort algorithm. There are two sets an open and closed set, i.e. points to which the final value respectively is not yet and is assigned. The values of the open set are stored in buckets of increasing value. All points except the starting points, which are put in bucket with zero value, are put in a bucket with infinite cost. At each step the bucket with the lowest value is emptied and its members ,if not already, assigned to the closed set. Then the open connected nodes are updated if the value of the generating node plus the arc length between the open and generating node is lower than the previous assigned value. This is continued until all points are in the closed set.

Now lets have a look at the metric. The natural metric for a Euclidean distance measure is just the Euclidean distance, e.g. for a 3×3 metric 1 for a tower step and $\sqrt{2}$ for a single bishop step. Borgefors showed that this is not optimal and introduced the integer based chamfer metric in which the maximum error is minimised. Verwer [141] published unbiased chamfer values, unbiased on average as none of these metrics are isotropic. This algorithm is referred to as EDT. For comparisons in this thesis we will use the optimal chamfer coefficients as published by Verwer [141].

Danielsson [24] introduced the vector based distance transform (VDT). The algorithm uses a two component vector to describe the distance for each pixel to the closest boundary pixel. The original algorithm was a 2 pass algorithm. Later replaced by an ordered propagation algorithm in Ragnemalm [106]. The original VDT suffered from small errors in a small number of specific configurations of boundary pixels. These problems where cased by the local update scheme and were solved by later algorithms. See for raster based error free Euclidean distance transforms

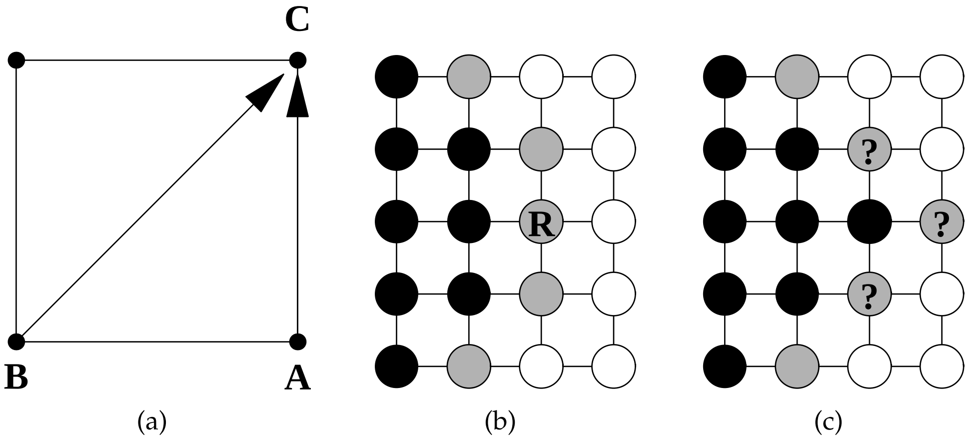


Figure 7.2: (a) Geometry of the grey value distance transform. (b-c) Dijkstra algorithm applied to an image. The black, grey and white grid points correspond to respectively set I, II and III. The point labelled R in (b) is the point in set II with the lowest cumulative cost. The points marked with a question mark in (c) are the points which have to be updated after the acceptation of R into set I.

e.g. Mullikin [94], Saito and Toriwaki [121] and for ordered propagation algorithms Vincent [144], Ragnemalm [106], Eggers [33], Cuisenaire and Macq [23]. These algorithms are out of the scope of this thesis and as such left as directions for the interested reader.

7.3 Grey weighted distance transform

The original Grey weighted distance transform was introduced by Rutovitz [117]. It was based on a two pass algorithm like the one in Rosenfeld and Pfaltz [116] in which first a pass is made from the upper left to the lower right followed by a pass in opposite direction. Piper and Granum [100] showed that the convergence properties of the GDT are comparable with that of the EDT in non convex regions, i.e. the 2 pass algorithm has to be iterated until convergence is reached Levi and Montanari [78]. The number of iterations then depends on the number of reversing paths. For sparse and complicated regions of interest a large number of iterations can be necessary. Therefore these sequential algorithms are not suited for the general case. As the sequential algorithm is dependent on the shape of the region of interest new propagation based algorithms were proposed. The propagation based algorithms process first points close to the initial front than points far away. An example is the recursive algorithm which after a point is updated recursively updates its neighbours until the entire recursive tree is processed. This limits the computational domain to the region of interest. By choosing an suitable order in which the neighbours are processed the performance can be increased. Although the recursive algorithm first processes points close by, it does not take into account the grey values. An algorithm which is generally more efficient than both the sequential and recursive approach is the ordered propagation method. In this type of method the subsequent points are processed based on their estimated arrivaltime. As such a front is propagated through

the entire image domain while processing only points close to the advancing front. As a result no iterations are needed and the image is traversed in an efficient way "minimising" the amount of pixel visits. The grey weighted distance transform is based on the Dijkstra algorithm, see Fig. 7.2 (b-c). Dijkstra [30] presented an efficient method to find the shortest path on a finite graph from some root branch to each other branch. This method returns a tree, i.e. a graph, with one and only one path between every two nodes. The method is order $O(nlogn)$ where n gives the number of nodes. The nodes are generated in increasing order, i.e. $O(n)$ as at each stage of the algorithm one node is added to the definite tree. The logarithm is due to the sorting process which decides which node is generated next. The algorithm utilises three sets. Set I contains all nodes definitely assigned to the tree under construction. Set II contains the nodes from which the next node is added to set I. Set III contains all nodes which are not yet considered or already rejected. The algorithm is initialised with all nodes but the root node put in set III where the root node is put in set II. For each added node to set I two steps are required. First the node with the shortest path length in set II is moved to set I and secondly the direct connected nodes are updated. If such a direct connected node is part of set I nothing is to be done. For a branch in set II the path length is updated if the value is lower than the old one. And finally if a branch is in set III it is moved to set II and the calculated path length is temporarily assigned to it. Only when a point is moved to set I its position in the tree is finalised and as such the path length is also definitive. The update process is repeated as long as set II contains nodes.
As for the Euclidean chamfer methods the propagation vectors are discrete and bounded to the principle axes and diagonals (for the 3×3 metric) the graph also immediately returns the lowest cost path from source to the point in question. For methods presented here after this is not true as the propagation vectors generally do not form a closed path from seed to the point of interest. As such a flow field approach is needed to return form reference point to seed.

Table 7.2: The 2 pass grey weighted distance transform on a 1D signal. The binary image contains the region of interest were the inverse serves as initial front. In the first pass the kernel is moved from left to right calculating the grey weighted cumulative sum from the closest seed on the left. The reverse pass updates the values of the first pass whenever the cumulative sum from the right is lower than that from the left.

binary image	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1
grey image	3	2	5	4	7	8	6	2	3	4	3	2	3	8	2
1^{th} pass	*	*	*	0	7	15	21	0	3	7	10	12	15	0	2
2^{nd} pass	10	7	5	0	7	14	6	0	3	7	8	5	3	0	2

7.4 Fast marching: a plane wave front approach (FMP)

In the mid 1990's the basis idea of the fastmarching algorithm was independently proposed by Tsitsiklis [136] and Sethian [127]. The main improvement with respect to the GDT lies in the modelling of the wavefront as a planar wave. The propagation

```
definition:
· $I$ cost image
· $U$ arrival time image
· $S$ set of points which constitute the initial front
· $P$ set of points forming a narrowband consisting of already touched points but
  which are still ahead of the front
· $A$ accepted set, points behind the wavefront
· $c_i$ the Chamfer weights for the neighbours $\boldsymbol{n}_i$ of $\boldsymbol{x}_{front}$
initialisation:
· $U(\boldsymbol{x}) \leftarrow 0 \ \forall \boldsymbol{x} \in S$
· $U(\boldsymbol{x}) \leftarrow \infty \forall \boldsymbol{x} \notin S$
· $P \leftarrow S$
loop:
  $while P \neq \varnothing$
    $\boldsymbol{x}_{front} \leftarrow \underset{x \in P}{argmin}(U(\boldsymbol{x}))$
    $for \ \{i | \boldsymbol{n}_i \notin A\}$
      $U(\boldsymbol{n}_i) \leftarrow min(U(\boldsymbol{x}_{front}) + c_i I(\boldsymbol{n}_i), U(\boldsymbol{n}_i))$
      $P \leftarrow P \cup \{\boldsymbol{n}_i\}$
    $end$
    $P \leftarrow P \backslash \{\boldsymbol{x}_{front}\}$
    $A \leftarrow A \cup \{\boldsymbol{x}_{front}\}$
  $end$
```

Algorithm 7.1: Grey weighted distance transform

direction of the wavefront is free and determined by the data, i.e. the propagation direction is not constrained to some privileged propagation vectors as for the GDT. The minimisation of the global cost path is implemented with a similar Dijkstra [30] like algorithm as the GDT. Sethian also proposed to apply a minimum heap sort algorithm to extract the minimum node from the graph which is of the same complexity as the Quicksort algorithm used for the GDT but has a better worse case performance.

The Fast Marching algorithm utilises the Dijkstra algorithm to construct the wavefront in a downstream fashion, i.e. points just ahead of the wavefront are ordered and at each step the point with the lowest arrival time is fixed and the arrival times of the not yet fixed 8-connected neighbours are updated. The update values are obtained by constructing a planar wave to the underlying cost function and the already accepted neighbours. Further the method locally assumes the propagation speed constant and the minimum cost path to be straight. Now let us look how a point C is updated from the already fixed points A and B. Fig. 7.3 shows a planar wave arriving at grid point C under an angle ϕ. As the propagation speed is constant and the path is assumed to be locally straight the difference in arrivaltime U_{AC} between the plane wavefronts with arrivaltimes U_C and U_A is simply given by the product of the distance between the wavefronts and the propagation cost. Therefor

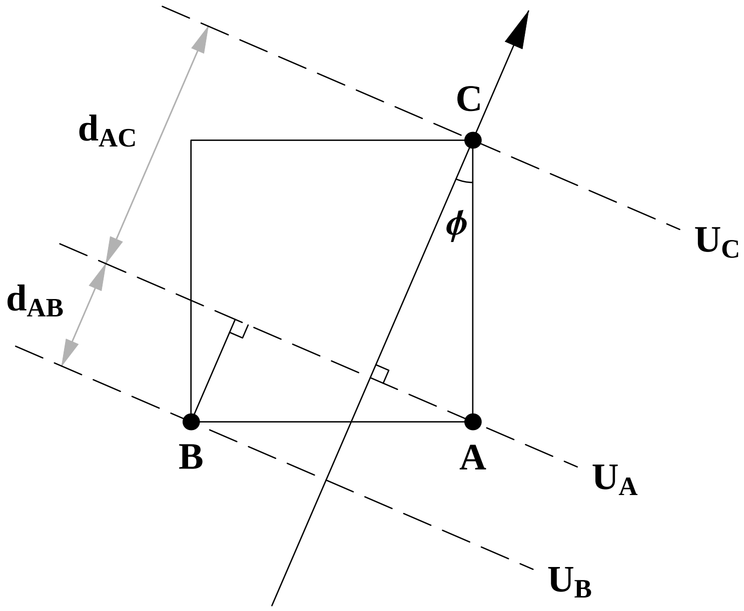


Figure 7.3: Geometry of a planar wave hitting grid point C under an angle ϕ. The dashed lines denote the wavefront passing through the points A, B and C with respective arrival times U_A, U_B and U_C. The distance between wavefronts U_A and U_B is denoted d_{AB}.

the arrivaltime of the wavefront at point C is given by:

$$U_C = U_A + \tau \, d_{AC} \tag{7.2}$$

where τ is the cost to travel a unit length through the grid cell and d_{AB} the distance between the wavefronts labelled U_A and U_B. Distance d_{AB} and d_{AC} are given by

$$d_{AB} = \sin\phi, \tag{7.3}$$
$$d_{AC} = \cos\phi. \tag{7.4}$$

Using the Pythagorean theorem or trigonometric identity, i.e. $\cos^2\phi + \sin^2\phi = 1$, The distance between the wavefronts passing through points A and C can be written as a function of d_{AB}

$$d_{AC} = \sqrt{1 - d_{AB}^2} \tag{7.5}$$

The cost to travel unit distance between the wavefronts U_A and U_B is given by

$$\begin{aligned} \tau_{AB} &= \frac{U_{AB}}{d_{AB}} \\ &= \frac{U_A - U_B}{\sin\phi}, \end{aligned} \tag{7.6}$$

where U_{AB} is the difference in arrival time between the respective wavefronts. In the planar approximation the cost function is flat therefore τ_{AB} in eq. 7.6 can be substituted by τ, the unit travel cost in the grid cell

$$d_{AB} = \frac{U_{AB}}{\tau} \tag{7.7}$$

combining eq. 7.7, eq. 7.5 and eq. 7.2 gives the arrivaltime of the wavefront passing through point C along the minimum cost path through the grid cell

$$\begin{aligned} U_C &= U_A + \tau\sqrt{1 - \left(\frac{U_{AB}}{\tau}\right)^2} \\ &= U_A + \sqrt{\tau^2 - U_{AB}^2} \end{aligned} \tag{7.8}$$

From this derivation based on the geometrical model it is not immediately clear why this should be a minimum cost path. However realizing that the solution is unique it should also be the minimum cost path. For the minimisation approach see e.g. Lin [79]. Basically there are 8 ways to update the point C in this setup. In the original algorithms presented by Sethian [127] and Tsitsiklis [136] updates of the arrival time were only initiated by 4-connected neighbours. Although the latter already proposed the 8-connected version, where the 4 connected algorithm is faster but less accurate. Both methods were evaluated in Lin [79].
An outline of the 8 connected fastmarching algorithm is given in Algorithm 7.2. The update function Υ is given by

$$\Upsilon(A,B,C) = \begin{cases} U_B + \tau & \text{for } U_B \leq U_A \\ U_A + \sqrt{2}\tau & \text{for } \tau \leq \sqrt{2}(U_B - U_A) \\ U_B + \sqrt{\tau^2 - (U_B - U_A)^2} & \text{else} \end{cases} \tag{7.9}$$

where τ is the cost associated with the octant $\triangle ABC$. Which can be defined in different ways, e.g. the average of the gridcell or octant. The different cases in this update function originate from the restriction that a point C updated from points A and B should be approached with an angle $\phi \in [0, \frac{\pi}{4}]$, i.e. the path should originate from the line segment AB. The cases from top to bottom respectively correspond to a solution approaching point C with an angle $\phi \leq 0$, $\phi \in [0, \frac{\pi}{4}]$ and $\phi \geq \frac{\pi}{4}$. Note that there's clipping of the angles for solutions originating from outside the octant. This is allowed as long as the cost function is continuous which is not always the case, i.e. when the cost in the octant is calculated as the average of the surrounding grid cell a point is possibly already surrounded by 4 different cost values. To make the Fast Marching better suited to cope with non flat cost functions one could think off fitting a plane through the cost plane e.g. writing τ as a function of position in the grid cell, e.g. $\tau(s,t) = \tau_C + s(\tau_C - \tau_A) + t(\tau_A - \tau_B)$. And then minimising the sum, $U_A + \int_0^{cos\phi} \tau(\zeta cos\phi, \zeta sin\phi)\, d\zeta$. Unfortunately no solutions of practical use were found although the cost function would be continuous.

7.5 Fast marching: spherical wave front (FMS)

Solving the Eikonal equations with the traditional FMP method is well suited for a wavefront which is not strongly curved. On the other hand in the case of a strongly curved wavefront the planar approximation fails and introduces significant errors. In this paragraph a better suited variant of the FM algorithm is presented which uses a circular approximation instead of a planar approximation to model the wave front. The presented solution is not in closed form although the solution is unique and can

```
definition:
· $I$ cost image
· $U$ arrival time image
· $S$ set of points which constitute the initial front
· $P$ set of points forming the narrowband consisting of already touched points but
  which are still ahead of the front
· $A$ accepted set, points behind the wavefront
· $\boldsymbol{n}_i$ are the 4 connected neighbours of $\boldsymbol{x}_{front}$
· Define $\boldsymbol{x}_{ij}$ as the $j^{th}$ element of $\{\boldsymbol{n}_j|(\boldsymbol{n}_i - \boldsymbol{x}_{front})\perp(\boldsymbol{n}_j - \boldsymbol{x}_{front})\}$ for given $\boldsymbol{n}_i$
initialisation:
· $U_x \leftarrow 0\ \forall \boldsymbol{x} \in S$
· $U_x \leftarrow \infty \forall \boldsymbol{x} \notin S$
· $P \leftarrow S$
loop:
  $while P \neq \emptyset$
    $\boldsymbol{x}_{front} \leftarrow \underset{x \in P}{argmin}(U_x)$
    $for\ \ \{i|\boldsymbol{n}_i \notin A\}$
      $for\ \ j \in \{1,2\}$
        $U_{n_i} \leftarrow min(\Upsilon(\boldsymbol{x}_{front}, \boldsymbol{x}_{ij}, \boldsymbol{n}_i), U_{n_i})$
        $P \leftarrow P \cup \{\boldsymbol{n_i}\}$
      $end$
    $end$
    $P \leftarrow P \backslash \{\boldsymbol{x}_{front}\}$
    $A \leftarrow A \cup \{\boldsymbol{x}_{front}\}$
  $end$
```

Algorithm 7.2: Fast Marching: plane wave approximation

be found quickly by simple root finding algorithms.
As for the plane wave approximation the arrivaltime at point C is given by the sum of the arrival time of the spherical wave at point A with the integrated cost to get from wavefront A to C. Locally a straight path and constant cost is assumed, therefor the integrated cost is simply the distance d_{AC} multiplied with the average cost along AC.

$$U_C = U_A + \tau_{AC}\, d_{AC} \tag{7.10}$$

where the distance between the wavefronts passing through A and C is given by

$$\begin{aligned} d_{AC} &= d_{CF} - d_{AF} \\ &= R - d_{AF} \\ &= R - \sqrt{R^2 + 1 - 2\,R\,\cos\phi} \end{aligned} \tag{7.11}$$

with R the radius of curvature and ϕ the propagation direction of the wavefront at point C. Where d_{AF} is obtained by applying the law of the cosines. Note further that for large R the model converges to the planar model, i.e. the curvature of the

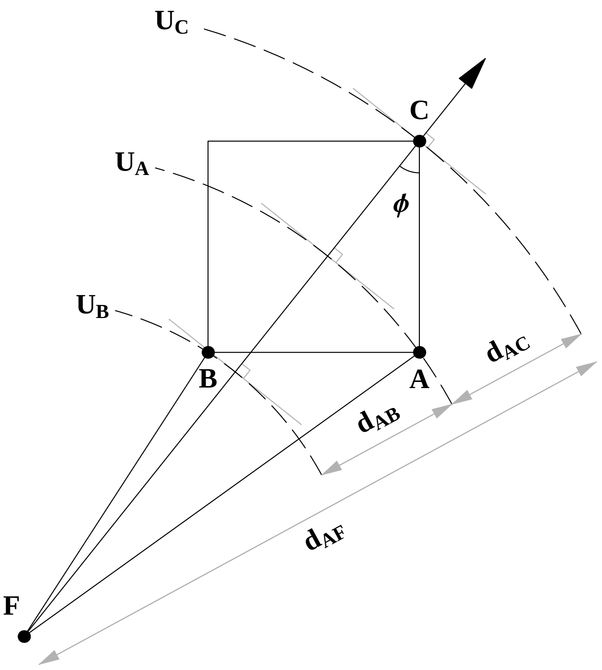


Figure 7.4: Geometry of a circular wave hitting grid point C under an angle ϕ. The dashed lines denote the wavefront passing through the points A, B and C with respective arrival times U_A, U_B and U_C. The distance between two wavefronts is denoted d_{XX}.

wavefront becomes negligible,

$$\underset{R\to\infty}{d_{AC}} = \cos\phi. \tag{7.12}$$

The distance between the wavefronts A and B with respective arrival times U_A and U_B is given by

$$\begin{aligned} d_{AB} &= \frac{U_A - U_B}{\tau_{AB}} \\ &= \frac{U_{AB}}{\tau_{AB}} \end{aligned} \tag{7.13}$$

with U_{AB} the difference in arrival time between the respective wavefronts. Independently of the underlying cost function τ_{AB} is the average of the cost at points A and B, i.e.

$$\tau_{AB} = \frac{I_A + I_B}{2} \tag{7.14}$$

From geometry on the other hand d_{AB} can be written as

$$\begin{aligned} d_{AB}(R,\phi) &= d_{AF} - d_{BF} \\ &= \sqrt{R^2 + 1 - 2\,R\,cos\phi} \\ &\quad - \sqrt{R^2 + 2 - 2\,R\,(cos\phi + sin\phi)} \end{aligned} \tag{7.15}$$

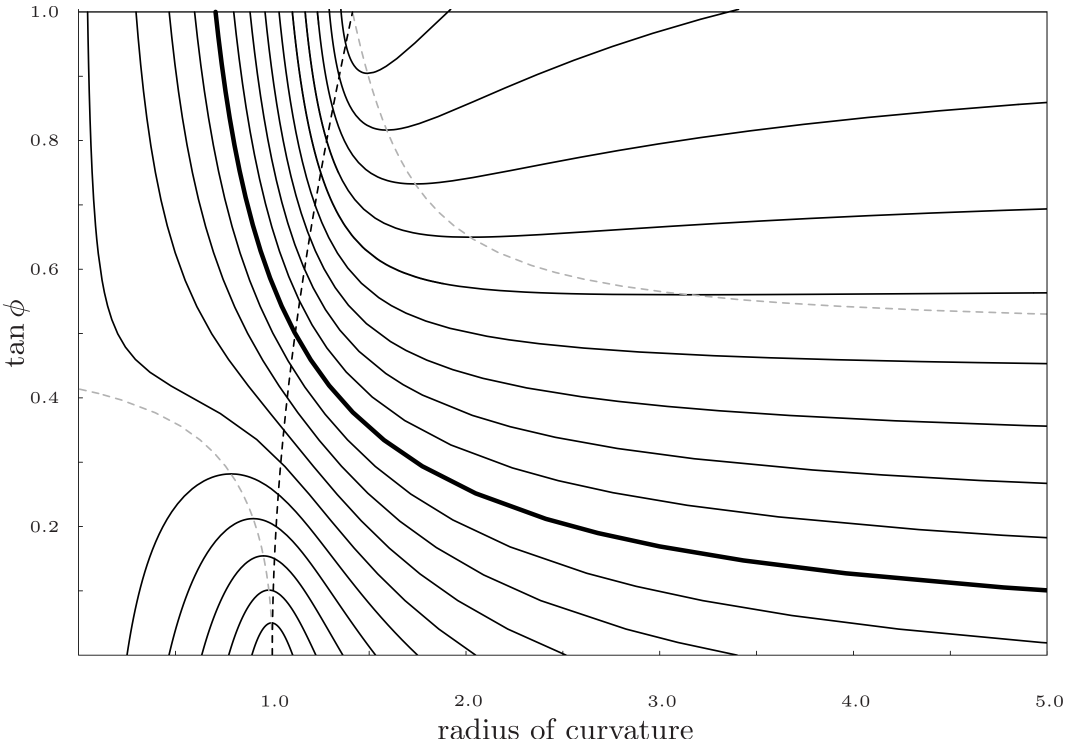


Figure 7.5: The function $R(d_{AB}, \phi)$ shown as the d_{AB} isolines. Following the cell boundary (black dashed line), the isovalue increases from -1 at $(R, \tan\phi) = (1, 0)$ to 1 at $(R, \tan\phi) = (\sqrt{2}, 1)$ with an 0.1 increment where the bold line is the $d_{AB} = 0$ isoline. Valid focus points, i.e. points outside the grid cell, can be found on the right of the black dashed line. At the grey dashed lines the isolines are horizontal.

Solving the set of equations formed by Eq. 7.13 and Eq. 7.15 reduces the solution of the spherical wave model for given d_{AB} to a function in R and phi. This function is plotted in Fig. 7.5 in the form of iso d_{AB} lines. The region of interest is confined to focal points outside the grid cell ($R \geq \cos^{-1}\phi$) which corresponds to the area on the right of the black dashed line and cost paths originating from within $\angle ACB$ ($0 \leq \phi \leq \frac{\pi}{4}$). As such the region on the left of the black dashed line is only of academic interest as all wavefronts represented by this region focus inside the grid cell. Further the figure shows that the function $\phi(R, d_{AB})$ is single valued in the region of interest.

Now we proceed by arguing that the radius of curvature R of the wavefront is only varying slowly. As such R can be replaced by an estimate $\tilde{R}$ based on the already constructed wavefront. If $\tilde{R}$ is assumed available, R in Eq. 7.15 can be replaced by $\tilde{R}$. As such the system of equations reduces to a function in ϕ. As no analytical solution was found an iterative approach was taken by means of the method of Newton (any other root finding algorithm would also do). From here on this iterative solution will be referred to as $\phi(R, d_{AB})$. Where the search range is given by $\phi \in [0, \frac{\pi}{4}]$. The radius of curvature $\tilde{R}$ can be deduced from the propagation directions of the parent points A and B. As we assume the wavefront to be diverging the lines through these points spanned by the respective propagation vectors will focus at some point F behind the current wavefront.

$$\begin{bmatrix} s \\ t \end{bmatrix} = \begin{bmatrix} \cos\psi_A & \cos\psi_B \\ \sin\psi_A & \sin\psi_B \end{bmatrix}^{-1} (A - B) \tag{7.16}$$

$$F = A - s \begin{bmatrix} \cos \psi_A \\ \sin \psi_A \end{bmatrix} \tag{7.17}$$

With ψ_X the propagation direction at point X. Then the radius of curvature, R, is approximated by the d_{CF}.

$$\tilde{R}(\psi_A, \psi_B; A, B, C) = \begin{cases} |F - C| & \text{if } \angle FAB \in [0, \frac{\pi}{2}] \wedge \angle FBA \in [0, \frac{3\pi}{4}] \\ \infty & \text{else} \end{cases} \tag{7.18}$$

the condition statement requires that the path is originating from $\angle ACB$. An alternative would be requiring some smoothness in the path and require that the angle between the FC from the estimate is within some range from propagation direction to be found in C.

Regularization

The propagation vector can be extracted from the method it self but when a bit more regularization is required it can be derived by applying normalised convolution to the known arrival time map with a suitable applicability function, e.g. Gaussian weighting. The certainty at a point is 1 if accepted and 0 if not. A sufficient basis is $\{1, x, y\}$. The coefficients for these basis functions supply the gradient vectors which coincide with the normal vector of the wavefront, see e.g. [75, 155, 159].
The radius of curvature can be found with a similar regularization. On one hand by calculating the isophote curvature on the known arrival time map with normalised convolution. Which is not trivial as due to the build up of the wave front kernels the certainty of the arrival time is zero in the Far set, unknown in the Near set and 1 in the Accepted set. Which means that the certainty is not randomly distributed over the kernel but localised at and behind the wavefront which means in certain cases that one cannot distinguish between higher and lower order kernels as such normalised convolution is only applicable as the basis functions are distinguishable which in general is not the case, i.e. for a 3×3 neighbourhood this already requires 7 points to be available with a reasonable certainty.
On the other hand the already proposed method based on the propagation directions can be generalised by incorporating a larger neighbourhood and a weighting based on the distance from the wavefront at A.

Symmetry breaking

In the special case a wavefront is constructed on the bottom of a ditch, where the centre line of this ditch is parallel to the grid and in the middle of two rows or columns, the algorithm behaves in a strange way. Note that in this case the sampled cost function is flat in all grid cells centered on the centre line also the arrival times U_A and U_B are equal. As a consequence the wave will behave as a spherical wave on a flat cost function, i.e. the radius of curvature will increase along the centreline of the ditch in the same way as it would along a radial line in the case of a flat costfunction with a single seed. This behaviour is caused by the modelling of the cost function by a plane spanned by I_A, I_B and I_C. To avoid this behaviour the sampling symmetry has to be broken. This can be done by either rotating the local coordinate grid or space. The solution at point C is needed therefore space has to be rotated. Therefor U and I are rotated. To this objective a rotation of the grid is applied to fit the wavefront after which the solution is rotated back to it's original position. This assumes that the interpolation used for the rotation uses sufficiently high order terms to describe the valley. Therefor a second order polynomial is fitted in a least square sense onto the cost function in a 4×4 neighbourhood centered on the grid cell. Further a Gaussian weighting is applied as function of the distance to the centre of the grid cell and a unity standard deviation.

The angle of rotation is the angle for which the bottom of a fitted parabola through the points I_A I_B and I_D falls on the line BC, where $D = A - AB$ with AB the vector from A to B. Where the parabola is extracted from the polynomial fit.

The polynomial fit allows to make a more accurate interpolation of the cost function. Therefor equation 7.10 can be replaced by

$$U_C = U_A + \int_0^{d_{AC}} p(\boldsymbol{x})\, dl \tag{7.19}$$

where p($\boldsymbol{x}$) is the second order polynomial fit with C as origin of the grid. Also the arrival time has to be adjusted in this case. Which is simply done by correcting the arrivaltime to the focal point over a distance to the focal point F. The correction is only applied for $d_{CF} \approx 5$. Not for larger values as the wave becomes planar and the correction infinite. A tapering would be better but would still involve an additional parameter.

Initialisation

For algorithm 7.3 the propagation directions for the initial front are required, i.e. in the spherical wave approximation the Neumann and Dirichlet boundary conditions are required simultaneous. Which is a problem as for the initial set the propagation directions can be multi valued [*], e.g. an isolated seed acts as an isotropic source and as such the propagation direction at this point seen from all the 8 connected neighbours is different. To avoid such ambiguities a new set is created in which all points on the boundary have a single propagation direction. This is achieved by defining a new image domain which is the 8 dilated set of the principle initialisation set. The GDT is applied on this set with as initial front the principle set, then the cost and propagation vectors of the difference set are used as initialisation of algorithm 7.3. Note that the arrival time on the principal set is constant therefor the propagation direction in the connected cells has to be perpendicular to this initial front. For single points this results in the propagation vectors of the GDT. On a elongated front the points can be described as single sources and as such the GDT still gives the correct description under the assumption that the cost in the cell is constant. As metric weights for the GDT the values 1 and $\sqrt{2}$ are to be used and not chamfer weights. Note that the GDT can not be replaced by the FM algorithm as it will result in errors

---

[*] in the imagedomain this can also happen in case of converging wavefronts

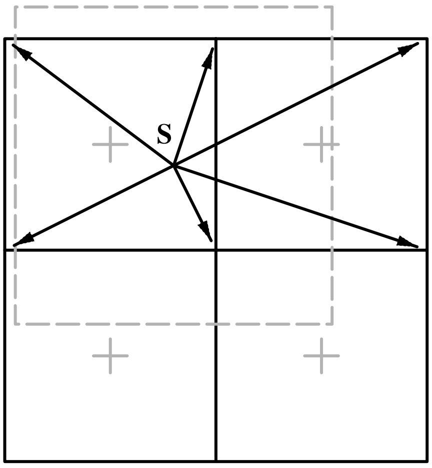


Figure 7.6: Construction of the propagation directions of the initial set for a single sub pixel seed. Points from grid cells with their centre (grey crosses) within 0.75 pixel ,in city block distance (dashed square), of the seed point, S are added to the initialisation set. The propagation vectors in these points are given by the respective difference vectors (arrows).

in the arrival time for 8 connected points. For example the cumulated cost to travel from a isolated source point to a 8 connected point through a grid cell with unity cost is for the FM method $1 + \frac{1}{2}$ instead of the true value which is $\sqrt{2}$.

For a single seed on the other hand we can start the algorithm with sub pixel accuracy. To this end the grid points which make up the surrounding grid cell are initialised with appropriate values. The propagation vectors are given by the difference vectors of the subpixel point with the surrounding grid points. The arrival time is obtained by the product of the length of the difference vector with the cost in the grid cell. When the subpixel point is situated on the boundary of a grid cell or at exactly at a grid point, all the points in the touching grid cells are to be included in the above initialisation. In practice however to make the initialisation better posed for points close to the gridcell boundary more grid cells can be added to the initialisation. This is accomplished by adding grid cells whenever the city block distance between the subpixel starting point and a grid cell centre is smaller than $\frac{3}{4}$. Otherwise stated when the border of a grid cell is closer than $\frac{1}{4}$ pixelwidth to the subpixel seed point it is also used in the initialisation procedure. Note that the difference vectors can span more than one cell. Therefor the arrival time at the grid points is calculated as a cumulative weighted sum over the cells, where the local speed is weighted with the distance travelled trough a specific grid cell.

A result of this pre initialisation step is that the algorithm is not initialised with a initial front but with a set of points which are connected and have some arrival time assigned to them. The latter should be kept in mind as in special cases the value assigned to points in this seed set can be higher then the arrival time in the region of interest.

```
definition:
· $I$ cost image
· $U$ arrival time image
· $\phi$ angle of incidence image
· $S$ set of points which constitute the initial front
· $P$ set of points forming the narrowband consisting of already touched points but
  which are still ahead of the front
· $A$ accepted set, points behind the wavefront
· $\boldsymbol{n}_i$ are the 4 connected neighbours of $\boldsymbol{x}_{front}$
· Define $\boldsymbol{x}_{ij}$ as the $j^{th}$ element of $\{\boldsymbol{n}_j | (\boldsymbol{n}_i - \boldsymbol{x}_{front}) \perp (\boldsymbol{n}_j - \boldsymbol{x}_{front})\}$ for given $\boldsymbol{n}_i$
· $\tau$ is the cost in a grid cell
initialisation:
· $U_x \leftarrow \begin{cases} U_x & \forall \boldsymbol{x} \in S \\ \infty & \forall \boldsymbol{x} \notin S \end{cases}$
· $\psi_x \leftarrow \psi_x \, \forall \boldsymbol{x} \in S$
· $P \leftarrow S$
loop:
  $while\ P \neq \varnothing$
    $\boldsymbol{x}_{front} \leftarrow \underset{x \in P}{argmin}(U_x)$
    $for\ \{i | \boldsymbol{n}_i \notin A\}$
      $for\ j \in \{1, 2\}$
        $\tilde{R} \leftarrow \tilde{R}(\psi_A, \psi_B; A, B, C)$
        $d_{x_{n_i} x_{front}} = \tau^{-1} (U_{x_{n_i}} - U_{x_{front}})$
        $\phi_{tmp} \leftarrow \phi(\tilde{R}, d_{x_{n_i} x_{front}})$
        $U_{tmp} \leftarrow \tau \, d_{AC}(\tilde{R}, \phi_{tmp})$
        if($U_{tmp} \leq U_{n_i}$)
          $U_{n_i} \leftarrow U_{tmp}$
          $\Psi_{n_i} \leftarrow \Psi_{tmp}$
        $end$
      $end$
    $end$
    $P \leftarrow P \backslash \{\boldsymbol{x}_{front}\}$
    $A \leftarrow A \cup \{\boldsymbol{x}_{front}\}$
  $end$
```

Algorithm 7.3: Fast Marching: spherical wave approximation

7.6 Path Extraction

Now assume a arrival time map, U is obtained for a given cost function and given initial front. The minimum cost path connecting some point on the map, P_{end}, with a point on the initial front, P_{start}, can be obtained by propagating in the direction perpendicular to the iso arrival time contours [21]. This minimum cost path is always tangent to the gradient of the arrival time, i.e. by following the gradient of the arrival time map the minimum cost path can be obtained. In literature this type of descent, i.e. following the flow of a vector field, is also referred to as streamlines.
A simple example is shown in Fig. 7.6, subfigure 7.7(b) shows the cost function with superimposed the extracted path. Fig. 7.7(b) shows the isochrones of the arrivaltime of a wave with the white point as starting point. Super imposed on these isochrones is the unsigned gradient field, over which the stream line is calculated with the grey point as starting point and the white as end point.
A simple way to obtain a stream line is by the Euler-forward method in which the coordinates along the streamline are given by:

$$\begin{cases} \boldsymbol{x}_0 = P_{end} \\ \boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \Delta l * \nabla T(\boldsymbol{x}_n) \end{cases} . \tag{7.20}$$

with Δl the stepsize between the successive points and $\boldsymbol{x}_0$ the start point of the stream line. More sophisticated methods are e.g. the Heun method and the Runge-Kutta methods [71]. All these methods are prone to zigzagging if the vector field is not sufficiently smooth. The paths, i.e. streamlines, presented in this thesis are calculated with the build in functions of Matlab. The gradient fields are obtained from the arrival time map by means of Gaussian derivative operators, with sigma .9 [145]. The minimum cost path for a GDT generated arrival time map is however obtained differently as the discrete nature of the GDT often causes zigzagging in the streamlines. This zigzagging would result in an unreliable length and curvature estimation. Therefor in case of a GDT generated arrival time map the descent is implemented as a 8 connected descent, i.e. only kings moves are allowed in the descent. As such the statistics as presented in [151] apply. The disadvantage is that length measurements become unreliable for short separations of the begin and end point.

7.7 Comparison of the distance transforms

In this section the performace of the different presented distance transforms , i.e. GDT, FMP and FMS, is compared while applied to a "infinite" line, i.e. for a time invariant wavefront. The arrival time of the test function has a hyperbolic profile perpendicular to the orientation of the line:

$$T(x, y) = y + b\sqrt{1 + \frac{x^2}{a^2}} + y_0. \tag{7.21}$$

with the x coordinate along the line and the y coordinate orthogonal to it, y_0 determines the position of the $T = 0$ wave front on the x axis. The cost function is given

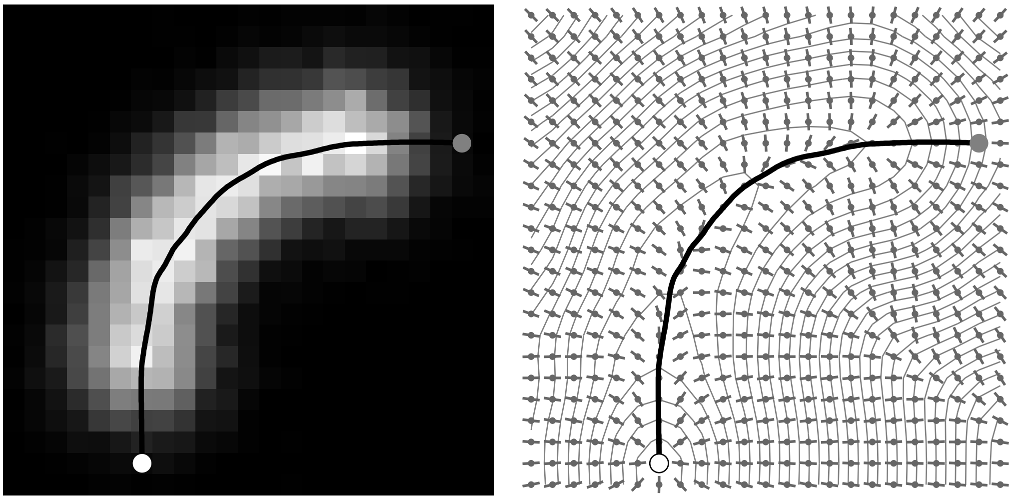


Figure 7.7: (a) A simple cost function (darker means a higher cost). (b) The arrival time for the costfunction in (a) with the white dot as seed is shown as contourplot where the time intervals between the contours are constant. The orientation of the gradient field is superimposed. Both figures further shown the streamline from the grey point to the white seed as calculated from the gradient field.

by the gradient magnitude of this function:

$$I = \sqrt{1 + \frac{b^2 + x^2}{a^4 + a^2 x^2}} \tag{7.22}$$

with $a > 0$ and $b > 0$. The opening angle of the front is given by $2\arctan(\frac{a}{b})$, see Fig. 7.8(b). The parameters can be expressed in the more intuitive units:

$$b = \frac{FWHM}{2}\sqrt{\frac{(m-1)(m+1)(1+3\,m)}{3+m}} \tag{7.23}$$

$$a = \frac{FWHM}{2}\sqrt{3 - \frac{8}{3+m}} \tag{7.24}$$

where $FWHM$ and m are respectively the full width at half maximum of the line and the suppremum of the cost function on a unbounded image domain. As such m defines the contrast between foreground and background as the foreground is put to unity. The function is analytic, tuneable and non flat i.e. there's a slight ramp on the sides of the line.

In this test a time invariant wave front is simulated and therefor the different algorithms should stick to this wave front while it is translated along the line. To avoid that the initialisation influences the result all points for which $T < T_0$ (Fig. 7.8(e) thick line) are initialised with the arrival time as prescribed by Eq. 7.21. In case of the FMS also the local propagation direction is given for $T < T_0$. In the example in Fig. 7.8(e) the FMS algorithm (solid line) sticks best to the true wavefront (dotted line) while the the isochrone of the FMP lags behind. The results of the GDT are not

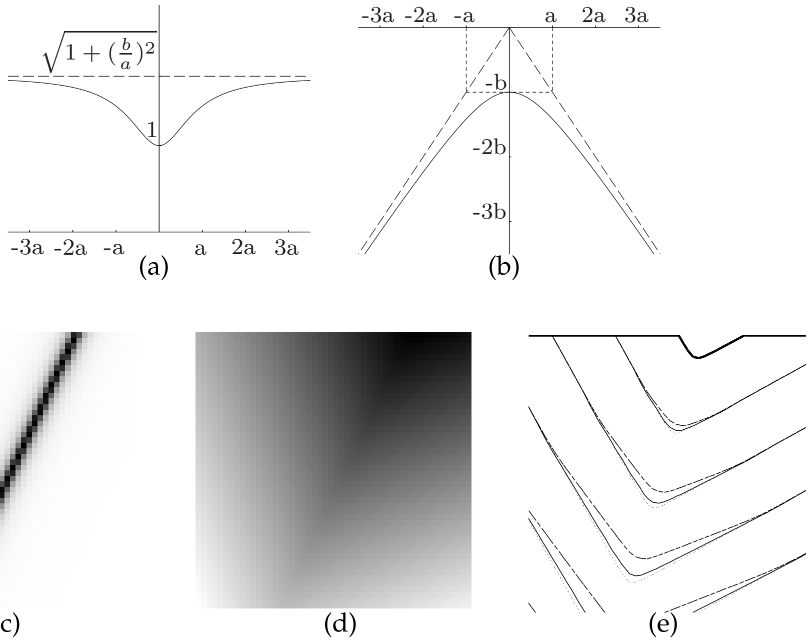
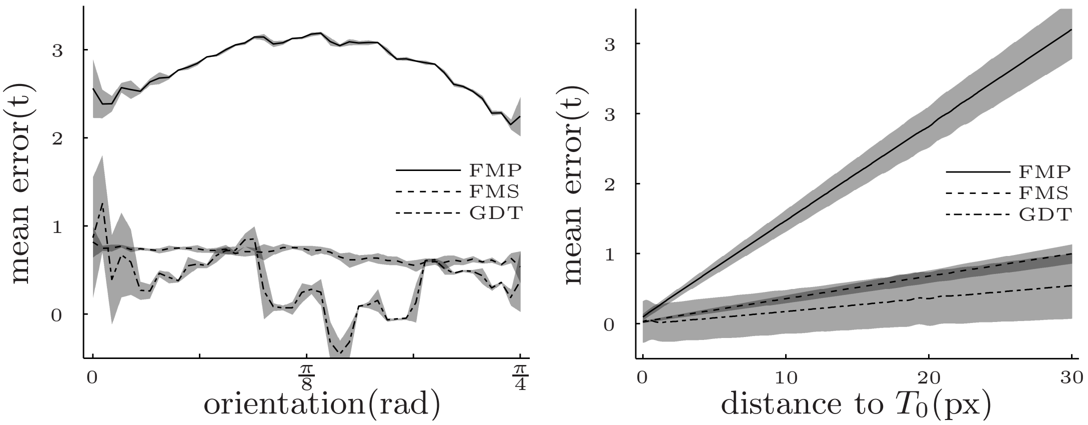


Figure 7.8: (a) Line profile, (b) isochrone, (c) cost function($a = 1, b = 1$), (d) arrivaltime for FMS on (c). (e) Isochrones, the tick line is the initialisation line, the dotted, dashed and thin solid lines correspond respectively to the ground truth and results obtained with the FMP and FMS algorithms. The isochrones are separated by equal time intervals.

shown as the spread in the results is quite large as will be shown in the next experiment.

Fig. 7.9(a) compares the angular dependence of the arrival time along random positioned lines at a distance of 20 px from the isochrone T_0. The standard deviation is shown as transparent grey patches. The GDT displays erratic behaviour as the GDT is made unbiased for lines under random orientation. Or stated otherwise the ratio between the number of moves, king and knight moves , is not constant along the line under arbitrary rotation and translation. As the translation for the GDT only results in a shifted version of the same "infinite" binary line for a single orientation little averaging occurs. The bias of the FMP is a much smoother function of orientation than the GDT. For the FM algorithms there's not a clear cut explanation as more variables enter the equation. The FMP shows some clipping artifacts due to the relative course interpolation of the cost function. In other words as the cost is assumed constant in a grid cell the cost is overestimated on edges. This overestimation can result in clipping which also can be observed for the FMP. As the cost is taken as the average in a grid cell it is almost always overestimated on non flat cost functions and as such explains the relative large error. Further more as the wavefront is assumed planar it also introduces an overestimate for curved wavefronts (under the assumption that the propagation direction is found correctly). The bias in the FMS algorithm is almost angular independent the bias is assumed to originate from the fact that the interpolating function is not fitting exactly to the hyperbolic profile and slightly overestimates the cost function.

Fig. 7.9(b) compares the arrivaltime along randomly oriented and positioned lines.



(a) (b)

Figure 7.9: (a)Mean error in the arrival time on the symmetry axis of a 'hyperbolic' line as function of the orientation of the line at a distance of 20 px from the initial front (FWHM=2 px, m=2), (b) same as in (a) but as function of the distance from the initial front, i.e. the lines are random oriented. In both cases the position of the lines is random.

The time is measured from isochrone T_0. The mean error at 20 px corresponds with the orientation averaged error in Fig. 7.9(a). As could be expected from the angular dependence the FMS shows the smallest spread followed by the FMP and then GDT. Concluding in regions in which the wavefront and or the cost function is strongly curved the error of the FMS method is almost independent of orientation and as such is much more suited to be made unbiased in particular cases in which the bias can be estimated.

7.8 Inner track

Path extraction based on distance transforms suffers from one mayor drawback, the paths extracted tend to take the inner bend for curved structures. This is just like a racing car driver will take the inner bend to minimise the time needed to get from start to finish. At the hart of this problem lies the Eikonal equation which minimises the cumulative cost or time needed to travel along the path, exactly like an athlete would do. A cartographer on the other hand would like to obtain the centre line of the road which basically minimises the average cost along the line instead of the time needed to travel along the path. So basically the inner track problem is only a problem if on uses it for a type of path extraction for which it was not designed, i.e. when then optimisation criterion differs from the one used. From our view point it is a problem as our interest was triggered by the question if we could extract the centre line of some DNA molecules in STM electron microscope images, see Chapter 9.

To investigate how severe this inner track problem is we introduce a cost function with a Gaussian profile:

$$I(d;\sigma,c) = 1 + c\,(1 - \exp^{-\frac{d^2}{2\sigma^2}}) \tag{7.25}$$

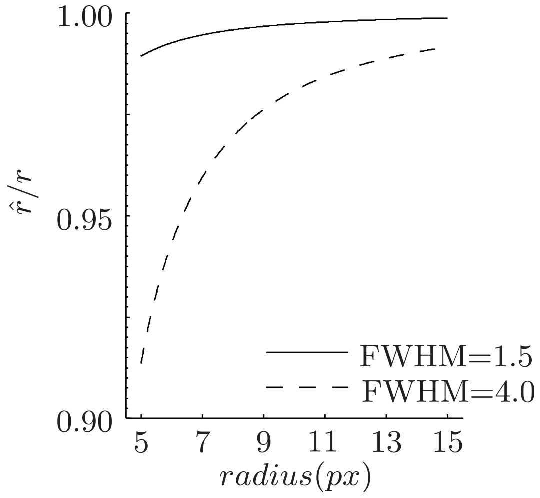


Figure 7.10: Relative radius of the true minimal circular cost path as function of the radius of the radius of the circular structure.

with d the distance to the centre line of the structure and c the contrast between foreground and background with the foreground put at unity. Now the full width at half maximum of the edge, $FWHM$, is related to the standard deviation of the Gaussian profile, σ, as

$$FWHM = 2\sqrt{2\,ln2}\,\sigma \tag{7.26}$$

To find the exact solution of the Eikonal equation we numerically solve the following equation:

$$\hat{r} = argmin(2\pi\hat{r}\,I(d;\sigma,c)) \tag{7.27}$$

with $\hat{r}$ the radius of the circular arch which minimises the cumulative cost along the circle and d the shift inward of $\hat{r}$ with respect to r. Where r is the true radius of the structure and as such $\hat{r} = r + d$ with $d \leq 0$. Now the relative radius is defined as the quotient of r and $\hat{r}$. Fig. 7.10 shows this relative radius found for a circular cost function with a cost profile described by Eq. 7.25 and $c = 3$. As can been from the figure the relative error is largest for small radii and decreases for increasing radii, i.e. when the arches become more line like and the inner track problem vanishes as could be expected. For very small radii the error drops again to become 0 for $r = 0$ but this regime is not visualised as on circular arches the profiles of the opposite points on the circle influence each other for a arc radius smaller than $\approx 4\sigma$. The broadest line $FWHM = 4.0$ obviously gives the worst results. Which can be explained by the observation that the gradient of the profile for this value is lower on the bottom of the profile and such the minimum cost path will move further inwards than in the $FWHM = 1.5$ case. For intermediate line widths one can safely be interpolated between the shown lines to get a feeling for the error to be expected. These values where key values in the next experiment, i.e. for smaller widths the lines cannot be sampled correctly any more and for larger widths no further interesting changes were found.

The simplest approach to minimise the innertrack problem is to scale the cost function, I, in a non lineair way. For example with a power function, I^{α} with $\alpha > 1$. This in effect reduces the width of the line. Note that a gradient is required for this method to work.

In the extraction of the centre line of a colon in medical image processing the inner track problem is tackled by thresholding the image and then calculate the distance transform from the colon wall. After inverting the result of the distance transform it can be used as cost function for a grey weighted distance transform [27]. This method is only usefull if the accuracy needed is moderate. Further the thresholding procedure should be feasible and the line profile should be broad enough not to be bothered by the binary colon wall.

Another approach could be to adapt the extracted path instead of the cost function itself. The flawed extracted path is then taken as starting point. Points are allowed to move along the line which coincides with the normal to the original line. Now each point is moved in the direction in which locally the average cost drops. The points are moved until a stable solution is reached or a sufficient number of iterations is performed. To make this method robust to noise some regularization is probably needed to keep the path smooth. This can for example be done by the sampling of the cost function on a larger scale.

Another approach is by introducing a little tension in the path. The tension tries to make the path contract as such it is in effect a curvature reducing term and counter productive if one wants to minimise the inner track effect.

The last proposed method depends on the observation that for a path with low curvature the inner track problem is small. Therefor a new method is proposed in which the path is straightened to reduce the inner track problem, see Chapter 8. Therefor at least two extractions are necessary, each successive extraction performed is done on a deformed grid, where the deformation is based on the results of its predecessor. The deformation is done in such a way that the precursor path is straightened. This is done by subsampling the cost function at regular intervals along the path found in the precursor step. At these points along the path the profile of the cost function is extracted, perpendicular to the path and with sufficient width. This results in a new cost image with the length along the path of the precessing path at one axis and the signed distance to this path on the other axis. After a successive path extraction is performed the resulting distance to the previous path can be used to correct the previous path. The process can be repeated several times depending on the accuracy needed and the noise characteristics. Preliminary tests showed promissing results. The method does not model the line profile nor does it put constraints on the curvature of the path. On the other hand it assumes a gradient which pushes the straightenend pathoutwards, i.e. it will not work on structures with a line profile with flat regions.

7.9 Comparison of the paths extracted with distance transforms

In this paragraph the presented path extraction method is evaluated for the different distance transforms, i.e. GDT, FMP and FMS. In the comparison two view points will be adopted. The first view point centres on the Eikonal equation and as such focuses on the point that a path should be a minimum cost path. From second vantage

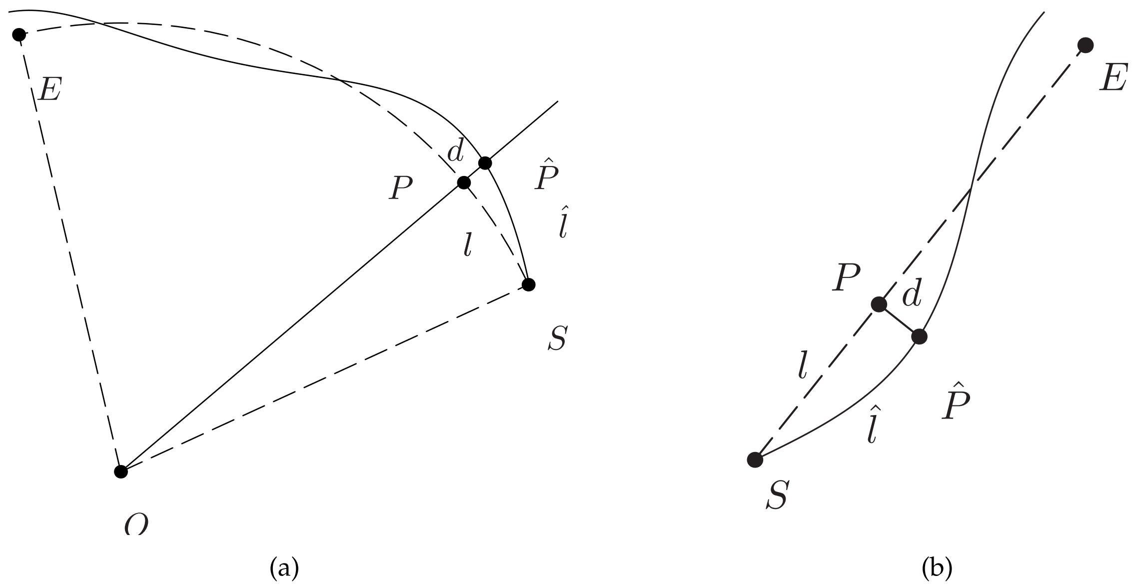


Figure 7.11: (a)Sketch of the experiment: the dashed line is a circular arc and serves as ground truth, the solid curve represents a constructed path. l and $\hat{l}$ denotes the path length along the respective curves (solid,dashed) from S to respectively P and $\hat{P}$ where $\hat{P}$ is the radial projection of P on the solid curve. d denotes the distance between P and $\hat{P}$. (b) Same as (a) but for a straight path.

point the extracted path should describe the underlying structure as good as possible. When adopting the second viewpoint one should realize that the algorithms are not designed to solve this problem and only converge in the limit when the curvature becomes negligible.

The tests are performed on circular arches and straight line structures. The line profile is given by Eq. 7.25 and the $FWHM$ for this profile is given by Eq. 7.26. For the circular arches and for the linear structures as well, c in Eq. 7.25 is put to 3. For an circular arch shortcuts can occur, in that case the path through the centre to a point on the opposite side of the arch has a lower cumulative cost than the path along the structure. Overflow will occur for $I_{max}\, 2\pi r \gtrsim I_{min}\, 2r$, assuming I_{max} is representative for the cost in the inner part of the circle and I_{min} is representative for the cost at the arch. As such the minimum value for $c \approx \frac{\pi}{2}$. This said, overflows are also prevented by masking out the centre of the circle in the tests.

Fig. 7.11(a) shows a sketch of the test for the circular arches. The distance transforms are started at point S while the streamlines are calculated from E to S. The dashed path denotes the true centre line of the grey arch with radius $r = d_{OP}$. Further $\hat{r} = d_{O\hat{P}}$ where $\hat{P}$ is the intersection point of the line d_{OP} with the extracted line. The length of the circular arch measured from S to P is denoted l while the distance over the extracted path between S and $\hat{P}$ is denoted $\hat{l}$. The arc length between S and P is for convenience reasons put to $\frac{2}{3}\pi\, r$. For the test the positions and orientations of the arches were randomised. For each point 250 realizations were used. As not all stream lines reached the start point the success rates for each noise realization is denoted in the graphs. In the following figures the standard deviation is denoted as hatched regions. Further note that the standard deviation is only calculated on the

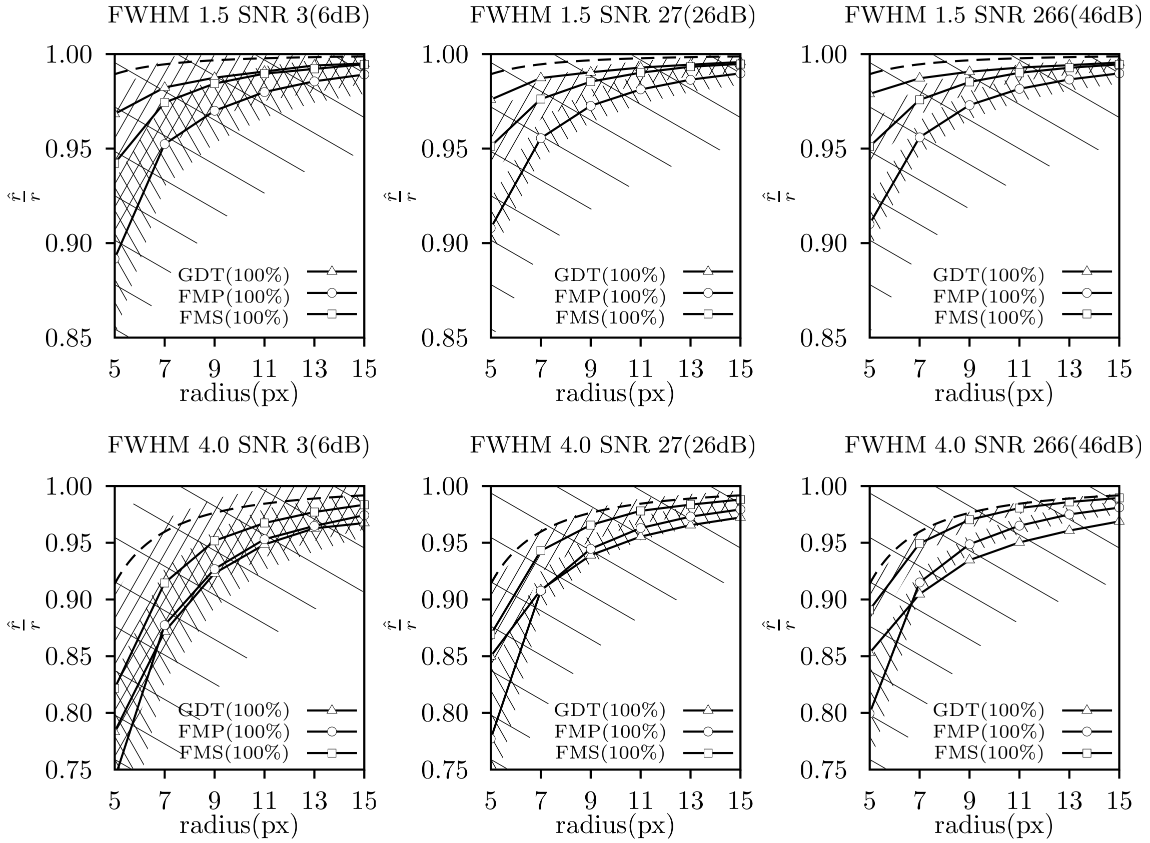


Figure 7.12: Measured relative radius for the GDT, FMP and FMS with the standard deviation denoted by the respective “⟍”,“\”and”/” hatched regions. The dashed line shows the true minimum cost path.

marked points and not in between.

The first Fig. 7.12 shows the relative radius found for the GDT, FMP and FMS. The dashed line denotes the relative radius for true minimum cost paths on a noise free cost function. The second Fig. 7.13 shows the relative radius of curvature which for a circular line should coincide with the relative radius. Here the radius of curvature is calculated as

$$r = \frac{1}{\kappa} = \frac{(x'^2 + y'^2)^{-\frac{3}{2}}}{x'y'' - y'x''} \tag{7.28}$$

where the derivatives in here are taken with respect to the arc length. Which can be easily derived from a cubic spline representation of the path, i.e. an piece wise polynomial which is differentiable to 2 order. A length comparison is not provided as is not distinguishable from a scaled version of the Fig. 7.12 with scalings factor π. Note further that the extracted paths were interpolated by a cubic spline interpolation to get the data at the disired points along the line [137, 86, 132].

Ideally the measurements would follow the dashed line in Fig. 7.12, clearly all algorithms tend to cut the corner with respect to the true minimum cost path. I some cases the GDT happens to perform best in terms of accuracy but this comes at the cost of precision loss. Further the standard deviation is almost an order of magnitude larger than that of the FMP and FMS. The improved performance of the FMS

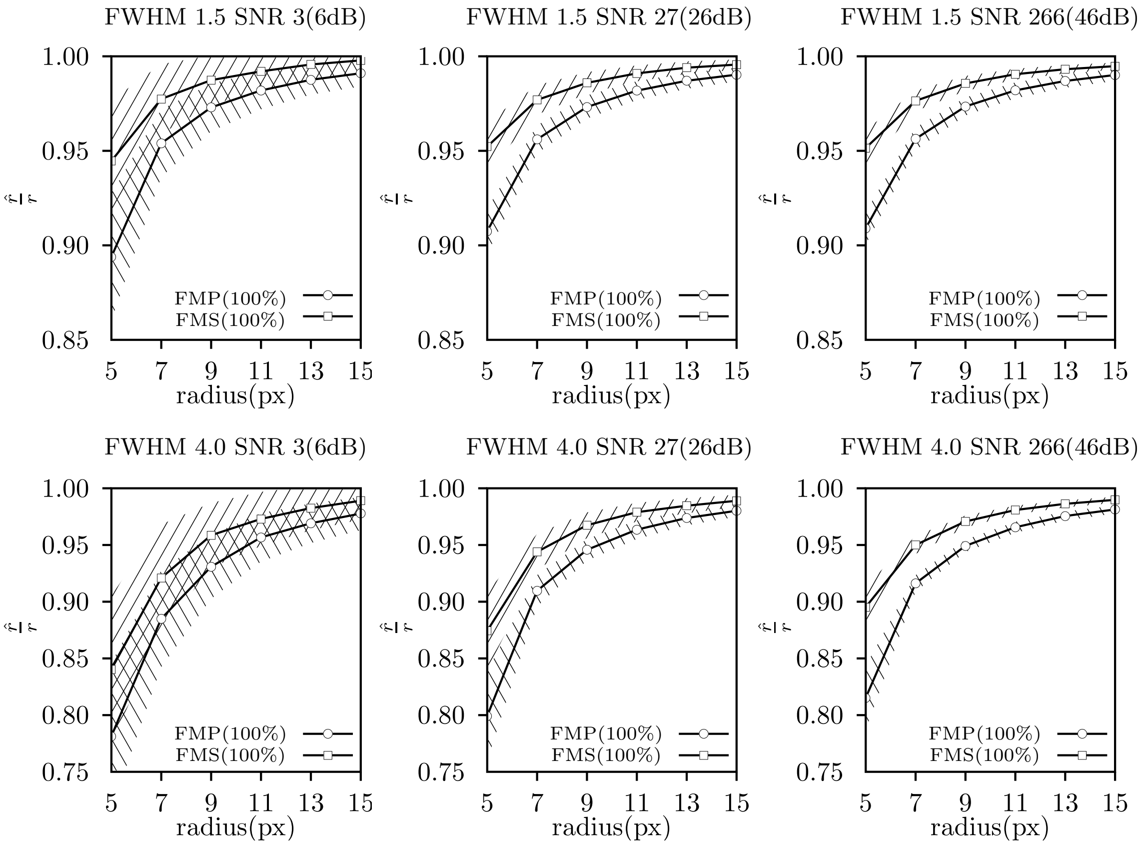


Figure 7.13: Measured relative curvature for the FMP and FMS with the standard deviation denoted by the respective "\"and"/" hatched regions.

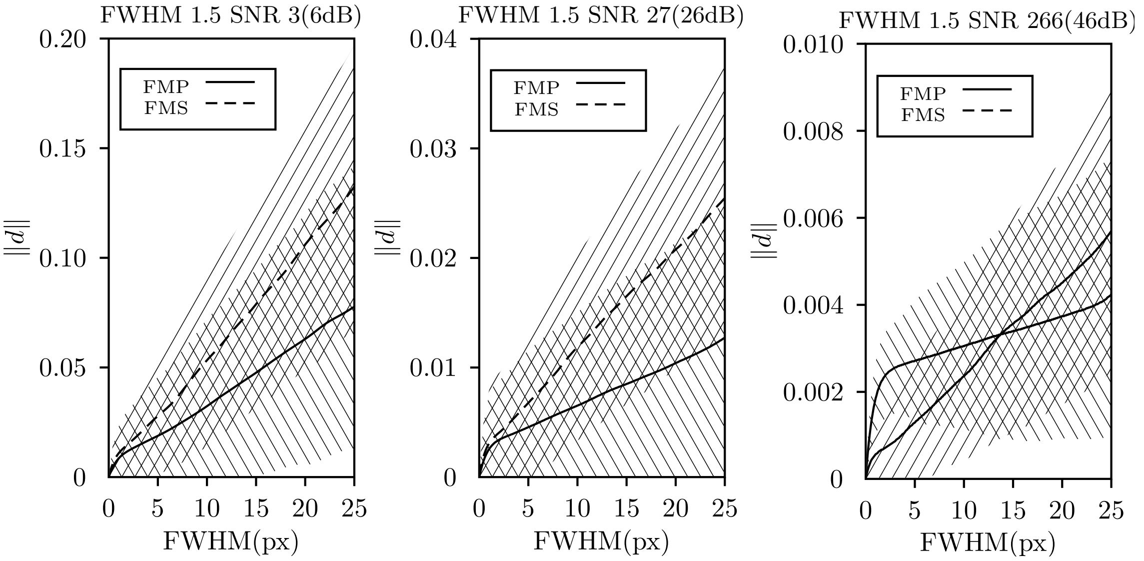


Figure 7.14: Difference in length between ground truth and measured values. The standard deviation is shown as hatched patches.

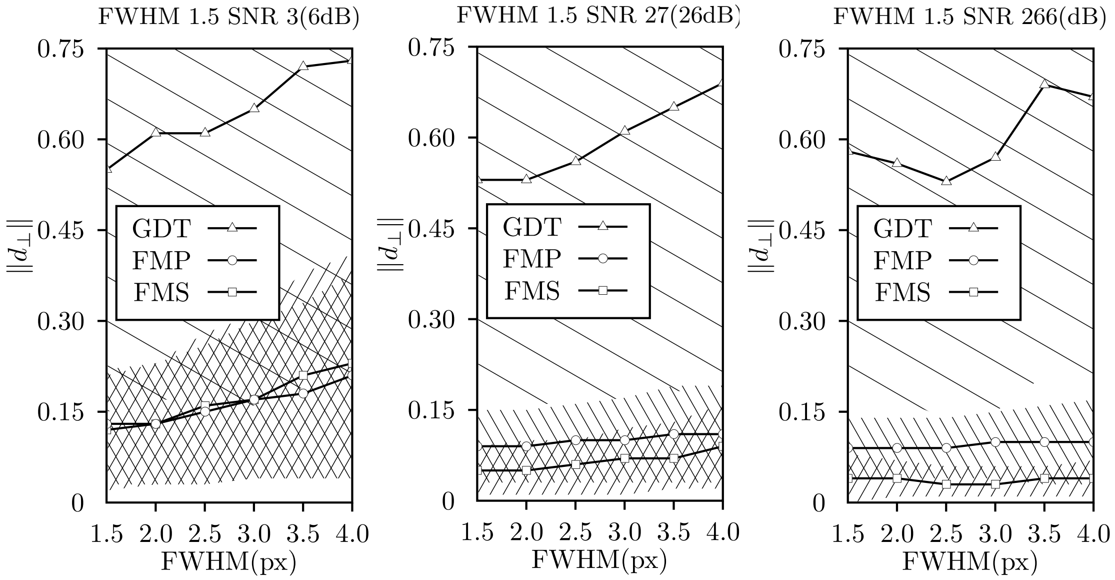


Figure 7.15: Shortest distance from the constructed path to the true line for the GDT, FMP and FMS. The standard deviation is shown as hatched patches.

over the FMP is contributed to the fact that the FMP models the cost function as a constant in the grid cell while the FMS fits a parabolic function.

Fig. 7.11(b) shows the straight line case of 7.11(a). The distance along the centre line and the extracted path are denote as respectively l and $\hat{l}$. While $\hat{P}$ is the intersection point of the normal of the centreline with the extracted path. The distance $d_{P\hat{P}}$ is denoted as d. Again the distance transforms are started from S and the streamlines are started at E and flow to S.

Fig. 7.15 shows the average absolute distance from the the true centre line for random oriented and positioned straight lines as function of the $FWHM$ of the line profile. Fig. 7.14 shows the length difference between $\hat{l}$ and l, i.e. the measured length and the true length along the path. The deviations are mainly caused by two mechanisms. First the extracted path is found parallel to the true path but slightly offset, this causes a relative large $|d|$-value and a larger errors at the start of a path while performing length measurements. The second reason is zigzagging which overestimates the length measurement but leaves the $|d|$ measurement relatively unharmed. From investigation of the individual paths the first is concluded to be dominating for the FMP based method. While the second explanation is dominant for the FMS based extraction. The zigzagging of the FMS can be contributed to the fact that it is a higher order algorithm that the FMP and as such is more sensitive to noise. The offset effect of the FMP on the other can be contributed to the sampling of the cost function which as the cost is averaged basically shifts the cost function. But as can seen for low noise cases the offset, $|d|$ and length measurements are for most applications fully acceptable, at least for the FM algorithms. For the GDT the length measurement are not shown as the standard deviation of the length measurements is about an order of magnitude higher than for the FM algorithms. The distance d for the GDT could be expected to be in the range, $[\frac{1}{2}, \frac{1}{2}\sqrt{2}]\,px$ as the average error along grid directions is of the order of half a pixel width while for a diagonal path this value increases to half the diagonal length, i.e. $\frac{1}{2}\sqrt{2}\,px$.

Finding the minimum-cost path without cutting corners

8

abstract[*]
Applying a minimum-cost path algorithm to find the path through the bottom of a curvilinear valley yields a biased path through the inside of a corner. DNA molecules, blood vessels, and neurite tracks are examples of string-like (network) structures, whose minimum-cost path is cutting through corners and is less flexible than the underlying centreline. Hence, the path is too short and its shape too stiff, which hampers quantitative analysis. We developed a method which solves this problem and results in a path whose distance to the true centreline is more than an order of magnitude smaller in areas of high curvature. We first compute an initial path. The principle behind our iterative algorithm is to deform the image space, using the current path in such a way that curved string-like objects are straightened before calculating a new path. A damping term in the deformation is needed to guarantee convergence of the method.

8.1 Introduction

Algorithms for computing the minimum-cost path have played an important role in various fields of science and engineering. These algorithms try to find the path connecting a selected start and end point that minimises the integrated costs. In optics, light rays travel along a minimum-cost path from source to destination. A wave front of light propagates with a speed that depends inversely proportional on the refractive index of the medium. A space varying velocity map suggests that the path with the shortest arrival time will in general be longer than the Euclidean distance between the start and end points. If you consider the local cost as the inverse of the local speed, then calculating the minimum integrated cost corresponds to calculating the smallest possible arrival time from a start point to all points in the domain.

---

[*] The content of this chapter has been published in: R. J. van Heekeren, F. G. A. Faas, and L. J. van Vliet. Finding the minimum-cost path without cutting corners. In B. K. Ersbøll and K. Steenstrup Pedersen, editors, *SCIA*, LNCS 4522, pages 263–272, June 10-14 2007 [59].

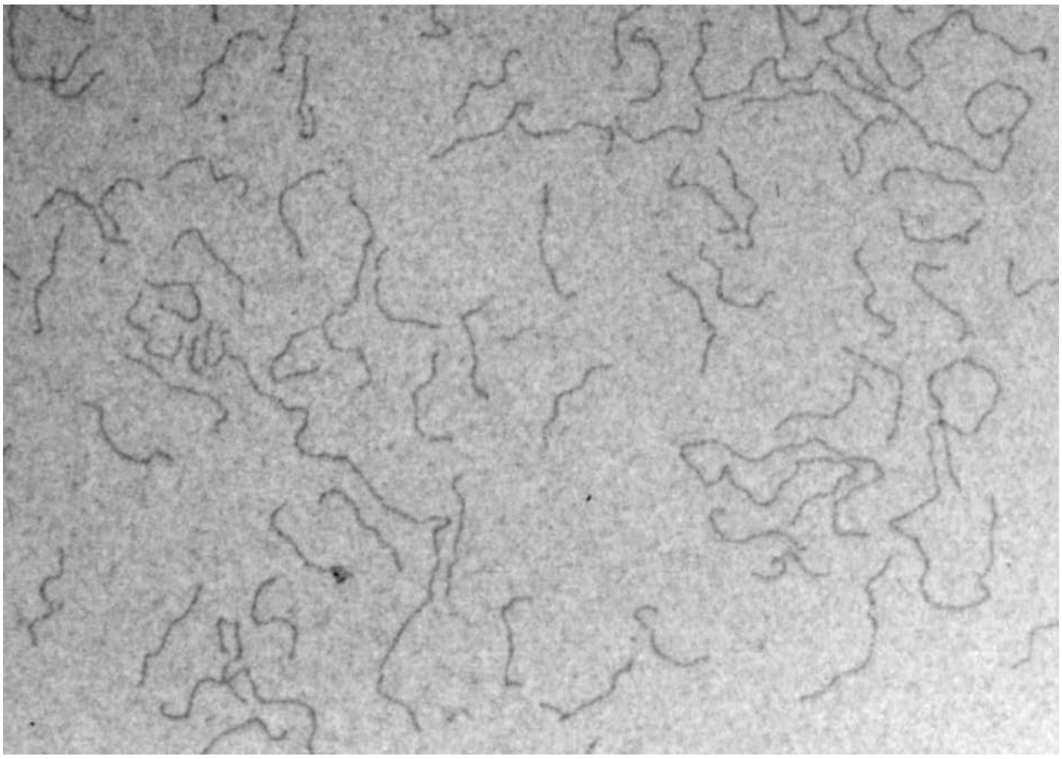


Figure 8.1: DNA molecules labelled with uranyl acetate and visualised by transmission electron microscopy. The images are kindly provided by Dr. Dmitry Cherny, PhD, Dr.Sc.

In many fields of science and engineering we encounter images of string-like structures in which the centreline conveys important information about the underlying objects. Examples are DNA-strands (cf. Fig. 8.1), blood vessels, or neurite tracks. The tracking results as depicted in [87] display exactly the problems that we are addressing in this paper. The minimum-cost path does not follow the curvilinear valley of the cost function, but is biased towards the inside of corners. In general, the minimum-cost path is cutting corners, and is therefore shorter and stiffer than the underlying centreline of the cost valley. Quantification of the bending energy of DNA plays a key role in understanding cellular processes. To verify the competing models and measure the so-called persistence length, an accurate path through these structures is required. The key to quantifying the length and shape of these objects is to find the centreline of these objects. A minimum-cost path guarantees a connected path that approximates this centreline even if the curvilinear object contains small gaps and is corrupted by noise (cf. Fig. 8.1). Solving the bias problem of minimum-cost path algorithms will be of utmost importance in many fields of science and engineering.

A typical implementation of such a standard minimum-cost path approach consists of the following steps:

- Define one start point in the image domain. Having an end point is not mandatory, but may assist in defining an early stopping criterion.
- Compute the minimum integrated cost from the starting point to all points in the domain, or until the pre-defined end point has been reached.

- Descend along the opposite gradient direction of the integrated cost image from the end point to the start point. Due to the smoothness of the integrated cost images one can obtain sub-pixel accuracy in the location of the minimum-cost path.

The minimum integrated cost T is given as the minimum cumulative cost along all possible paths $\boldsymbol{P}$ connecting the start point S with any end point E in our domain. Or mathematically:

$$T = \min_{\forall \boldsymbol{P}_{SE}} \int_S^E I(\boldsymbol{P}_{SE}(l))dl \tag{8.1}$$

with $\boldsymbol{P}_{SE}(l) = (x(l), y(l))$. This is equivalent to solving the Eikonal equation [13]

$$|\nabla u(x, y)| = I(x, y) \tag{8.2}$$

in which $I(x, y)$ denotes the local cost function and $u(x, y)$ the local arrival time. For uniform costs, the solution of the Eikonal equation is identical to the result given by the (domain constrained) Euclidean distance transform. For space variant costs we have the gray-weighted distance transform (GDT) [139, 133, 120, 43] and fast marching (FM) algorithms [3, 25, 136]. Both methods are based on wave front propagation. The GDT constructs a path using a superposition of cost-weighted basis vectors, thereby quantising the local path direction to the directions of a set of basis vectors in a 3x3 (or 5x5) neighbourhood. The FM algorithm models the wave front by a straight front, which does not restrict the propagation direction to a limited set of discrete directions. Both methods produce an image containing the minimum integrated cost from a starting point to all points in the domain.

The minimum-cost path can be obtained by a steepest descend (from the end point back to the start point) along the opposite gradient direction of the integrated cost map created by the aforementioned methods. Since the cost function is usually smooth, the integrated cost function is even smoother. This permits sub-pixel accuracy in computing the steps taken during the steepest descend. Due to the finite step size and approximation errors in the aforementioned algorithms, the path will not end exactly at the starting point but in very close (sub-pixel) proximity.

In section 8.2 we quantify the cutting corner problem for circular arcs with a Gaussian line profile and present our iterative algorithm to solve it. In section 8.3 we present quantitative results on the displacement error as a function of the number of iterations and qualitative results on TEM images of uranyl acetate labelled DNA. Section 8.4 presents the conclusions of our work.

8.2 Method

A correct implementation of a minimum-cost path algorithm applied to curved linear structures will always result in a path that is shorter and stiffer (less bending energy) than the centreline of the underlying linear structure. Especially in highly curved areas the minimum-cost path is cutting corners. The minimum-cost path does not follow the path through the minimum of the cost function in curved areas. To illustrate this we consider a circular path with a Gaussian cross-section

$$I(r, \sigma, c) = 1 + c\,(1 - exp(-\frac{(r - R_c)^2}{2\sigma^2})) \tag{8.3}$$

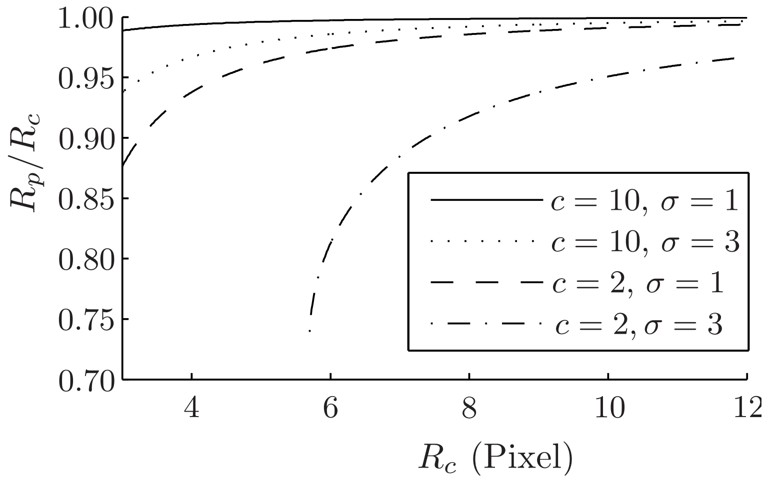


Figure 8.2: The relative radius (R_p/R_c) of the minimum-cost path for the Gaussian profile as a function of centreline radius R_c.

with the cost contrast $c = \frac{c_b - c_p}{c_p}$, in which c_b and c_p are respectively the cost values of the background and the path, and R_c the radius of curvature of the centreline. The integrated cost of a circular path with radius r around such a circle is

$$T(r) = 2\pi r\, I(r, \sigma, c) = 2\pi r\,(1 + c\,(1 - exp(\frac{-(r - R_c)^2}{2\sigma^2}))) \tag{8.4}$$

To find the minimum-cost path, we calculated the radius R_p for which $T(r)$ is minimised, $R_p = argmin\, T(r)$. Fig. 8.2 shows the relative radius R_p/R_c of the minimum-cost paths for different values of line width σ and contrast c. The results suggest that increasing the contrast or decreasing the line width (for example by scaling the cost function: $I(\boldsymbol{r}) \rightarrow I^{\alpha}(\boldsymbol{r}), \alpha > 1$)) of the cost function will reduce the bias in the minimum-cost path. In practice the bias will be reduced by these measures to some extent, but will never produce the desired smooth centreline path. This is shown by considering the limit ($c \rightarrow \infty$ or $\alpha \rightarrow \infty$), this will reduce the problem to a binary problem discarding all the gray value information and therefore produce a rough, binary skeleton type path instead of smooth centreline path. This skeleton path will also be hampered and possibly even interrupted due to the presence of noise in the original image. We claim to have developed an algorithm not based on increasing the contrast or decreasing the line width which solves the bias problem and still finds a smooth path, approximating the true centreline, through this class of objects.

Algorithm

Our method is based on the idea that a standard minimum-cost path algorithm such as FM will only give the correct centreline path for straight string-like objects (assuming the start and end points are located on the centreline). Hence, the principle behind our algorithm is to deform the image space in such a way that a curvilinear object becomes straight. After an initial path through the object is extracted using a standard minimum-cost path algorithm, two cubic splines are defined through the data points found by a steepest descend; one for the x-values and one for y-values

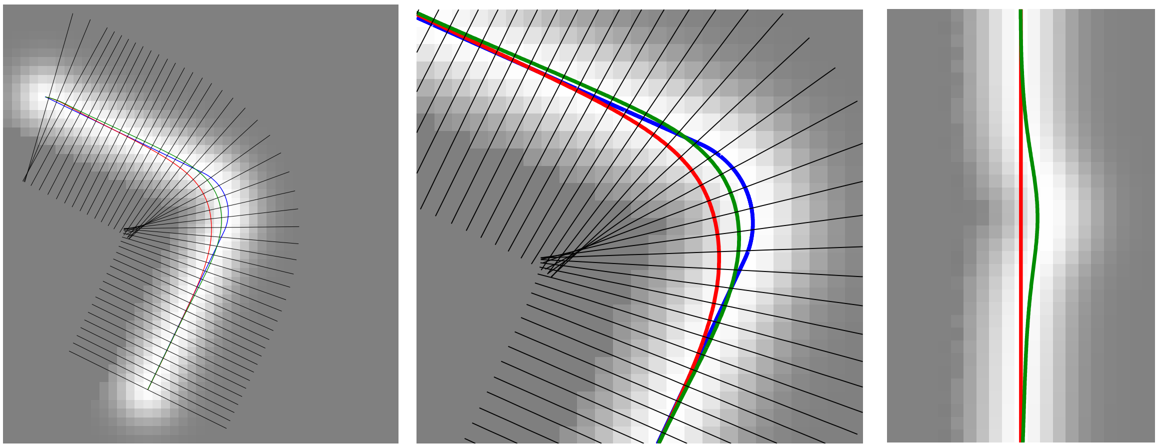

(a) (b) (c)

Figure 8.3: (a) The red line is the path extracted by a classical minimum-cost path algorithm, the green line is the path after the first iteration and the blue line is the true centreline of the object. (b) Zoomed in version of (a). (c) Deformed image obtained by equi-distant sampling perpendicular to the initial (red) path. The green path is the minimum cost path formed in the deformed image.

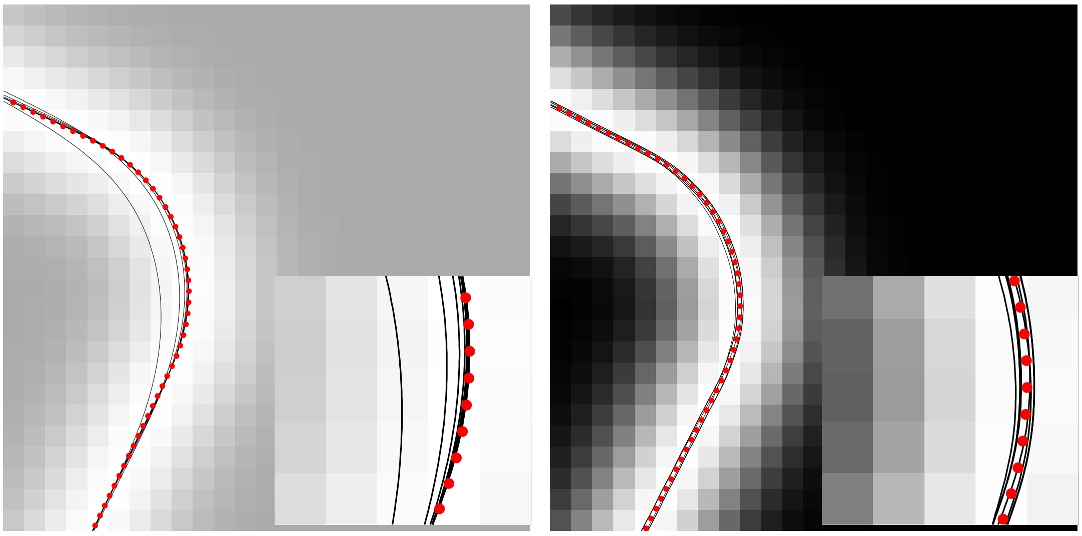
of the data points, using the distance from the end points along the path as the running variable. Using cubic splines guarantees that the path is continuous up to the second derivative. As shown in Fig. 8.3(a-b), lines perpendicular to the splines separated by a distance of one pixel are defined. A new image Fig. 8.3(c) is sampled using cubic interpolation on equi-distant points along these perpendicular lines. A new minimum-cost path is calculated in the deformed space (the green line in Fig. 8.3(c)). This new path is again represented by two splines. Next, the perpendicular distance between the centreline of the deformed image and the splines is calculated. By defining points on the perpendicular lines in the original image with the same distance from the original path, the new path is transferred back to the original image space. Two new splines are fitted through the coordinates of these points to produce a new path. As shown in Fig. 8.3(a), this path is already much closer to the desired centreline. Repeating the process described above yields a path that converges to the true centreline of the object.

8.3 Results

We first tested our algorithm on synthetic data, allowing us to measure its performance by comparing the results with a ground truth. Later we used images of DNA-strands made using an electron microscope to examine its real world performance qualitatively.

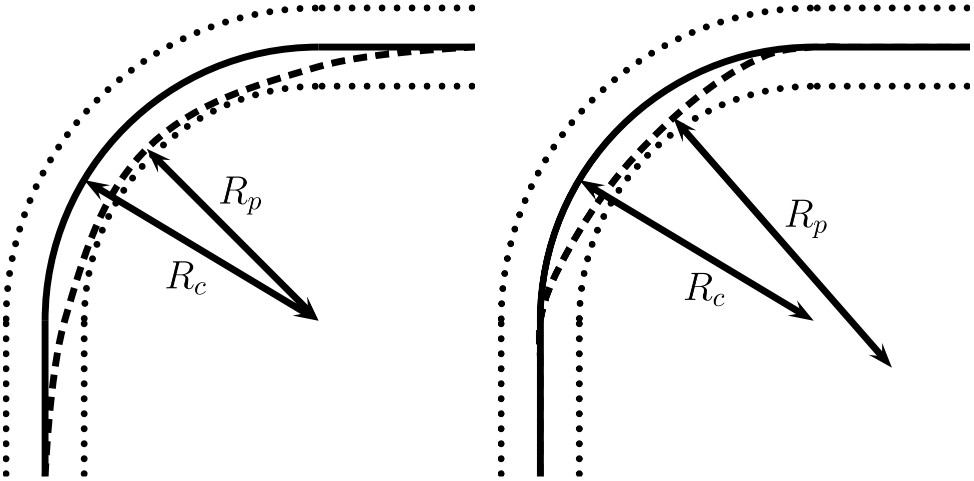
8.3.1 Synthetic Data

As synthetic data we used images of curved Gaussian line profiles, with a ninety degrees change in orientation and a curvature radius R_c (Fig. 8.3(a)). The cross section of this profile is defined as in Eq. 8.3. The algorithm was tested for different



(a) (b)

Figure 8.4: (a) A low contrast ($c = 1$) image depicting the path converging to the true centreline of the object. (b) A High contrast ($c = 10$) image depicting the oscillation effect causing the path to lie alternately on either side of the true centreline of the object. In both images the true centreline is denoted by the red dotted line.



(a) (b)

Figure 8.5: (a) The first way to cut a corner. Notice the radius of the path (R_p) being larger than the radius of the centrelineÂ (R_c). (b) The second way to cut a corner. Notice the radius of the path (R_p) being smaller than the radius of the centreline (R_c).

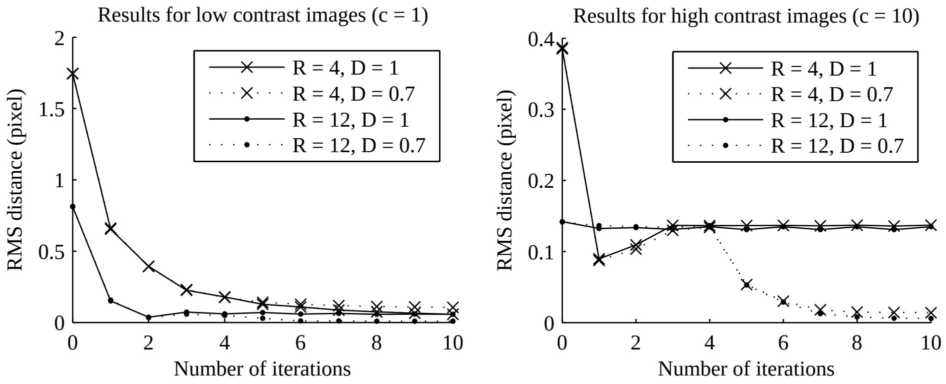


Figure 8.6: The mean of the RMS distance for twenty realizations as a function of the number of iterations using low ($c = 1$) and high ($c = 10$) contrast and line width $\sigma = 2.5$. For the cases with $D = 0.7$ the damping is switched on after four iterations.

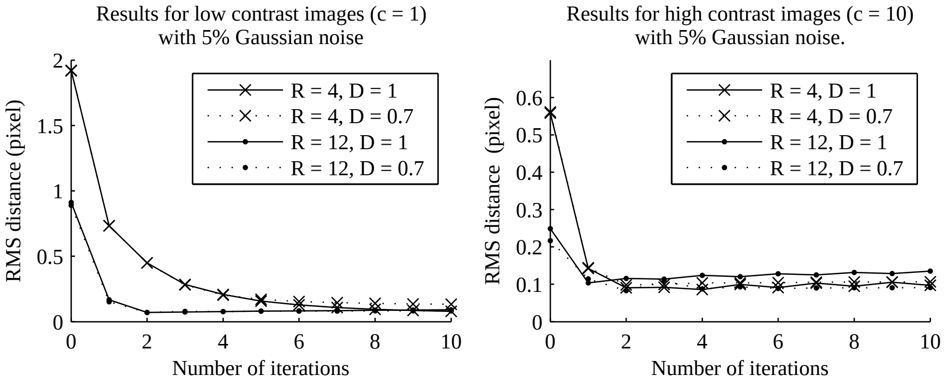


Figure 8.7: The mean of the RMS distance for twenty realizations as a function of the number of iterations using low ($c = 1$) and high ($c = 10$) contrast and line width $\sigma = 2.5$. 5% Gaussian noise is added to the images. For the cases with $D = 0.7$ the damping is switched on after four iterations.

centreline radii R_c, noise levels and contrast ratios c. As shown in Fig. 8.4(a) the first iteration already results in a path which is significantly closer to the centreline. We measured the performance by looking at the distance between the centreline of the object and the path found using our algorithm. We computed the root-mean-square (RMS) of the perpendicular distance between the path and the ground truth at ten points separated by a pixel in the middle of the curve.

Initially this RMS error decreases for all the settings. However, after a number of iterations (one to three for the high contrast images and about six for the low contrast images) it starts to increase for certain values of radius R_c and contrast level c. This is due to an oscillation effect, which results in the paths lying alternately on either side of the centreline of the object between successive iterations. The effect is depicted in the close up of Fig. 8.4(b). We suspect it originates from the two fundamentally different ways to cut a corner. In Fig. 8.5 the two possible ways are shown. On the left side the radius of the path R_p which cuts the corner is larger than the radius of the centreline R_c of the object. In contrast to the situation on the right where the radius of the path is smaller. Due to this sharper bend, we overcompensate for the bending and hence change sign of the curvature in the deformed space. In cases with a bend

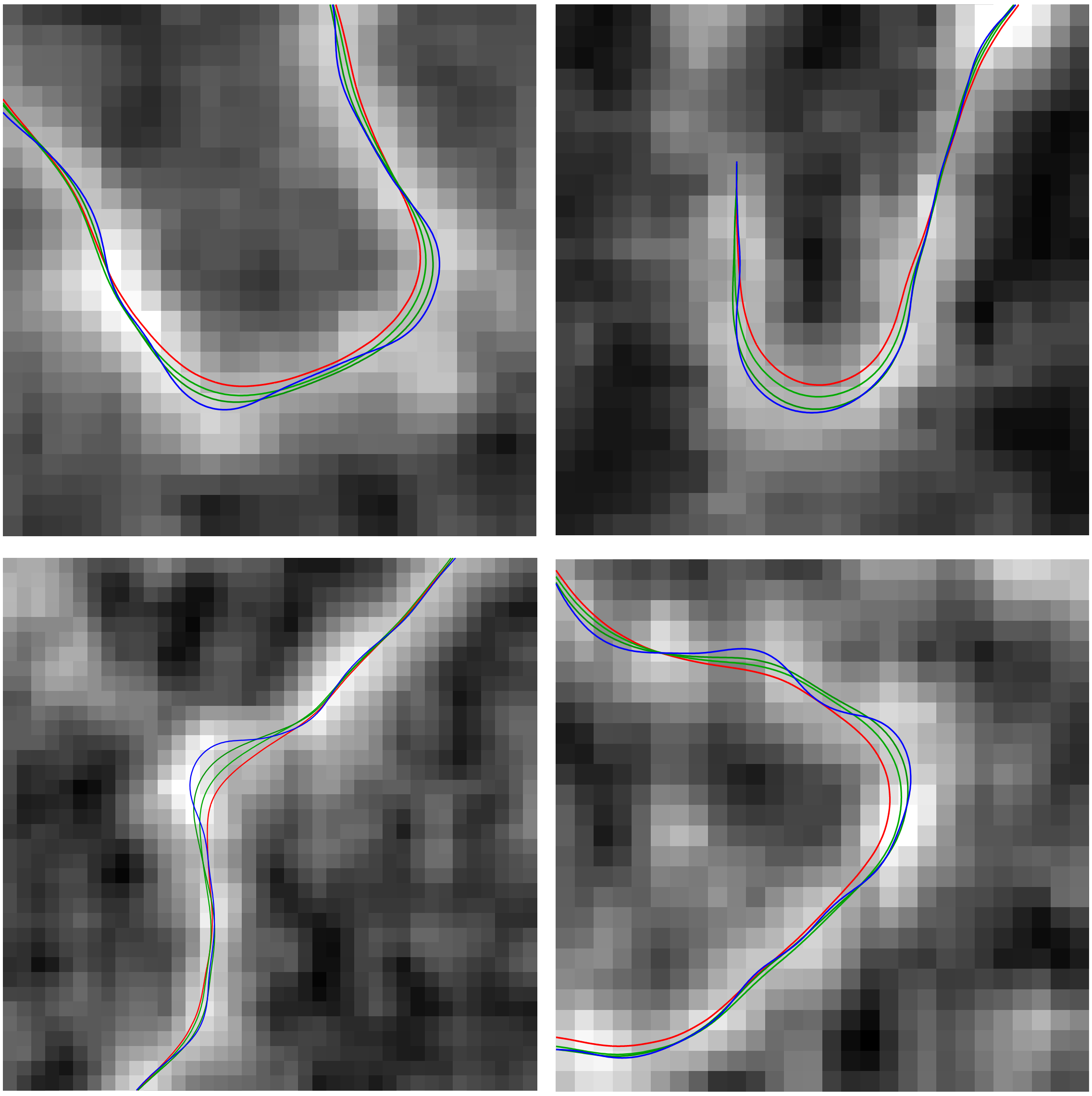


Figure 8.8: Four TEM images of DNA-strands. The red line depicts the path found by a standard minimum-cost path algorithm (FM). The green lines indicate the results of our method after one and four iterations, the blue line after 25 iterations.

which is less sharp than the true centreline, we only partially compensate and hence do not change the sign of the curvature. Therefore, the oscillating effect is only observed when an intermediate path has a sharper bend than the true centreline of the object and significantly cuts the corner.

To counteract this effect we introduced a damping term at the transition of the path from the deformed space to the original image space. The damping is being reduced exponentially. After N iterations the distance between the last and the new path in the n^{th} iteration is multiplied by a damping factor $D^{(n-N)}$ $(D < 1)$. This damping assures stable results. Elaborate testing has showed us that $D = 0.7$ is either the optimal or near optimal over a wide range of values for c and R_c. Only very low contrast settings require less damping to allow the path to reach the centreline.

In Fig. 8.6 the mean of the RMS distance of twenty realizations is plotted as a function of the number of iterations with low ($c = 1$) and high ($c = 10$) contrast settings

and no noise added. The plots show that the RMS error decreases dramatically in comparison with standard minimum-cost path algorithms (the 0^{th} iteration) for all radii and contrast levels. The damping is switched on after the fourth iteration. We observed that the damping decreased the RMS distance on all high contrast images, but on the low contrast images only for the curves with a large radius.
Fig. 8.7 shows the mean RMS distance for images with 5 percent Gaussian noise added. The RMS distance slightly increases after a number of iterations. This is not due to oscillating behaviour but caused by the fact that the path also adapt to the noise pattern. For medium to high SNR's the path corrections in the first iterations are dominated by the signal. The iterative procedure should stop when the noise becomes the dominant factor.

8.3.2 Real Data

The proposed algorithm has been extensively tested on transmission electron microscope images of DNA-strands labelled with uranyl acetate to quantify their shape. Empirically we deduced that twenty-five iterations were sufficient to reach a stable result on all of the images. Because no oscillating behaviour was observed, no exponential damping was used. Fig. 8.8 shows four typical results from the more than thousand molecules that were processed. The red line is depicting the path found by the fast marching algorithm, the blue indicates the final result after twenty-five iterations, the green lines in between are the results after respectively one and four iterations. Note that the final results describe the centreline of the object much better, especially in regions with high curvature. The blue line follows the local minimum of the cost function without cutting corners. This work permits the computation of the persistence length of DNA with much greater accuracy, especially over small distances. Earlier results always overestimated the persistence length in this regime due to the stiffness of the minimum-cost path.

8.3.3 Computational Speed

The time needed for one iteration is comparable to the time needed to calculate a classical minimal-cost path. Therefore, it is evident that the amount of computation needed increases approximately linearly with the number of iterations. Note that one often can limit the amount of image space to be evaluated after the first iteration, hence reducing the computation time in subsequent iterations.

8.4 Conclusion

In this paper we present an improvement on minimum-cost path algorithms, which significantly boosts their performance in describing the centreline of string-like objects. The method can be incorporated in any minimum-cost path algorithm. We have demonstrated that our algorithm results in a path which corresponds much better to the centreline of both simulated and real-world string-like objects. The RMS displacement error decreases more then a factor of ten, especially in highly curved areas. Displacement errors of several pixels can be repaired. The behaviour depends on conditions such as contrast, noise level and line width. Under certain conditions, such as high contrast, the method only converges after incorporating a damping term. Ten to twenty-five iterations are needed, using an exponentially reducing damping term after several iterations. The method has been successfully applied to several thousands of DNA molecules in high-resolution images obtained by TEM and AFM. The paths we obtained on the images of DNA-strands follow the valley through the cost function without cutting corners. Hence the length measurement remains unbiased and the curvature is no longer underestimated. This is of utmost importance for measuring the bending energy of DNA-strands on a nanometer scale.

DNA deformations near charged surfaces: electron and atomic force microscopy views

9

abstract[*]
DNA is a very important cell structural element which determines the level of expression of genes by virtue of its interaction with regulatory proteins. We use electron (EM) and atomic force microscopy (AFM) to characterize the flexibility of double-stranded DNA (about 150-950 nm long) close to a charged surface. Automated procedures for the extraction of DNA contours ($\sim$ 10-120 nm for EM data and $\sim$ 10-300 nm for AFM data) combined with new statistical chain descriptors indicate a uniquely two-dimensional equilibration of the molecules on the substrate surface regardless of the procedure of molecule mounting. However, in contrast to AFM, the EM mounting leads to a noticeable decrease in DNA persistence length together with decreased kurtosis. Analysis of local bending on short length scales (down to 6 nm in the EM study) shows that DNA flexibility behaves as predicted by the worm-like chain (WLC) model. We therefore argue that adhesion of DNA to a charged surface may lead to additional static bending (kinking) of $\sim$ 5 degrees per dinucleotide step without impairing the dynamic behavior of the DNA backbone. Implications of this finding are discussed.

9.1 Introduction

DNA's charge properties and its high flexibility allow irreversible adhering of molecules onto a planar surface forming a two-dimensional (2D) object and thereby permitting visualization by means of EM and AFM. In general, adhesion and immobilization are achieved by virtue of ionic interactions between DNA phosphates and surface charges. These processes lead to (i) a loss of one degree of freedom thereby dimin-

---

[*] The content of this chapter has been published in: F. G. A. Faas, B. Rieger, L. J. van Vliet and D. I. Cherny. DNA deformations near charged surfaces: Electron and atomic force microscopy views. Biophysical Journal 97(8):1148-1157, 2009. [36].

ishing the number of possible DNA configurations and (ii) potential distortions of the DNA double helix reflected in notable alterations in DNA conformation(s) which may lead to a change in the apparent persistence length. A quantitative description of the apparent DNA conformation confined to a plane requires invoking the existing models for DNA conformation in solution (3D) and in plane as well. The worm-like chain (WLC) model treats DNA as a continuous, inextensible (elastic) rod and considers the deformations occurring at each infinitesimal point following Hooke's law[76, 77]. Another model is based on a discrete description of a DNA polymer chain attributing its conformational flexibility to the thermal fluctuations in the angles between adjacent basepairs [122]. Similar to the WLC model, the discrete model describes DNA as a homopolymer. The main statistical quantities describing behavior of homopolymers by either model have similar analytical expressions allowing the use of either description for the analysis purposes. Several extensions were made towards a more realistic model, such as the inclusion of static bends or kinks distributed randomly (in position and orientation) along the polymer chain [123, 124, 102, 156, 157, 93].

Numerous examples demonstrate the power of EM and AFM in determination of DNA persistence length and the conformational state of DNA confined to an imaging plane [93, 48, 112, 113, 20, 97, 80, 158, 57, 83, 47, 138]. In many instances, the persistence length calculated close to 50 nm [48, 112, 113, 20, 97, 158, 2, 95], a generally accepted value determined by other techniques exploiting mainly bulk measurements [57, 83, 80]. However, smaller values (36 nm [17]) and larger values ($\sim$ 80-140 nm [93, 37, 162, 112]) were reported as well. In addition, there is disagreement about the state of the deposited molecules: (i) either molecules equilibrate on the imaging surface prior their adhesion (called 2D state), or (ii) molecules captured ("trapped") by the imaging surface without their equilibration leading to the conformation reflecting a projection of 3D conformation in solution onto 2D plane (called 3D state). A mixture of both, 2D and 3D, states can be present for each molecule or their ensemble depending on the deposition procedure and DNA length. Both, EM and AFM provide evidence(s) for a 2D state [48, 47, 112] and a 3D one [63, 112, 138]. In this line, recent studies show that DNA flexibility may vary in a length dependent manner exhibiting an increased flexibility (via spontaneous large-angle bends) over distances <5 nm [158]. Even the break-down of the conventional elastic rod model is suggested for short DNA fragments [85].

In this paper we will argue that successful analysis of 2D data sets requires both dedicated image processing and a model describing the statistical behavior of DNA molecules confined to a plane. We will show that DNA immobilized onto a surface can exhibit a notable variation in persistence length, yet showing its chain statistics as expected over long separation distances (in EM from 10 to 120 nm and in AFM from 10 to 300 nm) along the contour. By extending the number of statistical quantities we prove a 2D equilibrium state of immobilized molecules. A modification of the homopolymer WLC model was required to provide an adequate description of the DNA configuration in plane. To this end, we introduce local surface-induced static bends leading to an averaged heteropolymer WLC model [122, 123, 124, 20]. We also argue that modulations in apparent DNA persistence length due to interactions with the imaging surface are not a methodological artifact, but rather a general property of DNA that also manifests itself upon interaction with protein surfaces.

The developed software for the image and data analysis is provided as Matlab (The Mathworks, USA) scripts which can be freely downloaded from `www.diplib.org/home22266`. It provides the possibility to analyze images obtained from either EM or AFM imaging given sufficient image quality and provide DNA characterization as presented below. The software makes use of the Matlab toolbox DIPimage (TU Delft, The Netherlands, www.diplib.org).

9.2 Results

9.2.1 Image processing and analysis

Recently, we developed an automated procedure for finding a path through the centerline of string-like structures, such as DNA, obtained, for instance, by high resolution imaging techniques [59]. Briefly, the method (see Materials and Methods) is based on a minimum-cost path algorithm and allows determination of a path whose distance to the true centerline is more than an order of magnitude smaller in areas of high curvature than traditional algorithms. The method has been successfully applied to several thousands of DNA molecules on high-resolution images obtained by EM and AFM. The paths extracted by this method stay in the middle of the DNA strand without cutting corners. As a consequence, the length measurement remains unbiased and the curvature is no longer underestimated. Fig. 9.1 shows a typical EM micrograph of 474 bp long DNA fragment, an AFM image of a mixture of 474 and 2505 bp long DNA fragments and the convergence of DNA centerline (from red via green to blue).

The method was validated by Monte-Carlo simulations of DNA molecules following WLC statistics in plane (see Materials and Methods). Validation of the method allowed us to analyze not only full-length DNA fragments, but also short segments after a magnification calibration. We found that isolated and non-self intersecting strands behave identical to pieces of self-intersecting molecules, i.e. persistence lengths consistent within 1 nm (data not shown). This is very advantageous as longer molecules tend to intersect more and the chances of finding isolated molecules without self- intersection decreases.

Average statistical quantities were calculated as a function of length along the DNA based on all scored (without self-intersections) molecules as described in Materials and Methods after which we fitted by the corresponding formulae 9.4-9.6 to the measured quantities. Special precautions were made to avoid intrinsic correlations between measured quantities for different distances along the molecule. Figs. 9.2 and 9.3(a-c) exemplifies the results of estimations of persistence length calculated from average squared end-to-end distance (Eq.9.4), average squared angular difference (Eq.9.5) and average cosine of angular difference (Eq.9.6), respectively. The data show very good agreement with theoretical expectations and remarkable consistency of persistence length calculated using different formulas. The uncertainty in determining the persistence length due to the fitting was in the range of 1-2 nm. We applied this procedure to the data drawn from various experiments and found very good consistency among the fit results obtained from the different formulas (see Table 9.1). Together with the Monte-Carlo simulations, the results on the experimental data validate the procedure for image processing and data analysis.

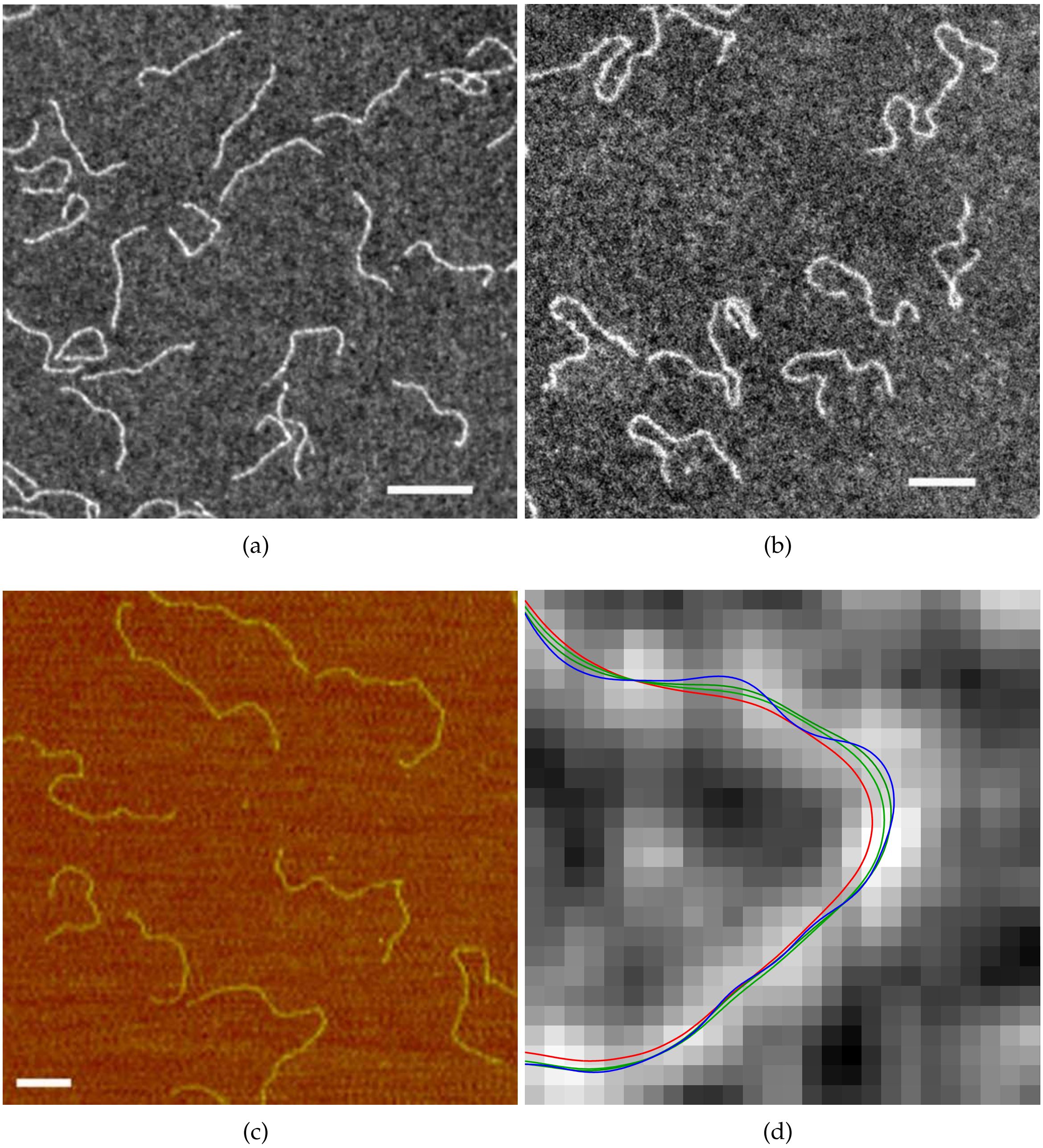


Figure 9.1: a) EM micrograph of 474 bp DNA fragment mounted onto polylysine film from 1 mM NaCl; b) and c) EM and AFM images of DNA digested with EcoRI and ScaI (1144bp and 1837bp fragments respectively), and mounted from 5 mM Hepes, pH 7.5, 10 mM KCl and 5 mM $MgAc_2$; d) cartoon showing convergence of automatic iterative image processing (from red via green to blue) as described in Materials and Methods to find the DNA backbone. Scale bars, 100 nm. Images a,b and d are displayed using inverted grayscale.

9.2.2 Dimensionality of surface immobilized DNA and its persistence length

The statistical quantities for the conformation description of homopolymers in a 3D are:

$$\begin{aligned}
\left\langle R^2 \right\rangle_{3D} &= 2P_{3D}^2\left(\frac{L}{P_{3D}} - 1 + e^{-\frac{L}{P_{3D}}}\right) , && (9.1)\\
\langle \cos\theta \rangle_{3D} &= e^{-\frac{L}{P_{3D}}} , && (9.2)\\
\left\langle \theta^2 \right\rangle_{3D} &= \frac{2L}{P_{3D}} , && (9.3)
\end{aligned}$$

where L is the length along the DNA (or contour length) between two points on a polymer segment, P_{3D} is the persistence length in 3D, θ is the angle between two tangent vectors separated by a distance L along the DNA. The formulae 9.1-9.3 used for the description of polymer chain statistics are similar for continuum (WLC) and discrete models [76, 77, 122, 123] and do not account for self-avoiding behavior. For the 2D case the corresponding formulae are [48, 112]

$$\begin{aligned}
\left\langle R^2 \right\rangle_{2D} &= 2(2P_{3D})^2\left(\frac{L}{2P_{3D}} - 1 + e^{-\frac{L}{2P_{3D}}}\right) , && (9.4)\\
\langle \cos\theta \rangle_{2D} &= e^{-\frac{L}{2P_{3D}}} , && (9.5)\\
\left\langle \theta^2 \right\rangle_{2D} &= \frac{L}{P_{3D}} . && (9.6)
\end{aligned}$$

Note that Eqs. 9.4-9.6 and Eqs.9.1-9.3 display identical mathematical structures. The only difference is a scaling of the persistence length between 2D and 3D, i.e. with the substitution $2P_{3D} \rightarrow P_{2D}$ in Eq. 9.1-9.3 the formulas are identical. In the remainder of this paper we will use $P_{2D} = 2P_{3D}$ when explicitly referring to the 2D model. In view of the above considerations, this implies that, in a strict sense, the persistence length can be determined with an uncertainty factor of two in absence of prior knowledge of the conformation dimensionality. To avoid this uncertainty, we computed the average fourth moment of the end-to-end distance for the molecules and compared the value with the model in 2D and 3D. For the WLC model the expression has been derived in 3D [103]

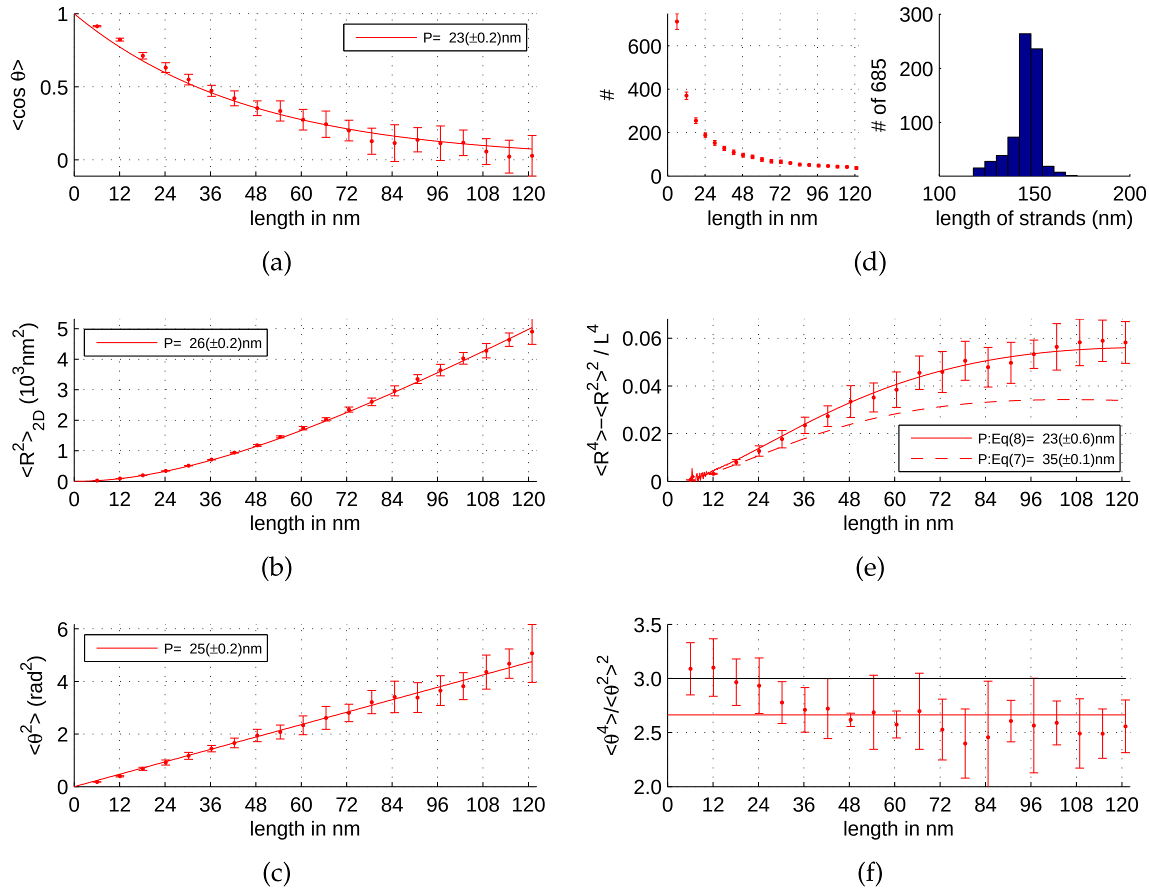
$$\begin{aligned}
\left\langle R^4 \right\rangle_{3D} = {} & \tfrac{20}{3}L^2P_{3D}^2 - \tfrac{208}{9}LP_{3D}^3 + \tfrac{856}{27}P_{3D}^4 - 8LP_{3D}^3e^{-\frac{L}{P_{3D}}}\\
& -32P_{3D}^4e^{-\frac{L}{P_{3D}}} + \tfrac{8}{27}P_{3D}^4e^{-\frac{3L}{P_{3D}}} . \qquad (9.7)
\end{aligned}$$

For 2D the expression is (see Eq. 9.29 and [129, Eq. C18])

$$\begin{aligned}
\left\langle R^4 \right\rangle_{2D} = {} & 32\,L^2P_{3D}^2 - 240\,LP_{3D}^3 + 696\,P_{3D}^4\\
& -\frac{320}{3}LP_{3D}^3e^{-\frac{L}{2P_{3D}}} - \frac{6272}{9}P_{3D}^4e^{-\frac{L}{2P_{3D}}} + \frac{8}{9}P_{3D}^4e^{-\frac{2L}{P_{3D}}} . \qquad (9.8)
\end{aligned}$$

Importantly, the formulae for $\langle R^4 \rangle$ do not scale from 2D to 3D. However, the difference vanishes for lengths exceeding ~ 500 bp, after which $P_{3D} = \sqrt{6/5}P_{2D}$. In contrast, the difference

$$\left\langle D^4 \right\rangle = \left(\left\langle R^4 \right\rangle - \left\langle R^2 \right\rangle^2\right)/L^4 \qquad (9.9)$$



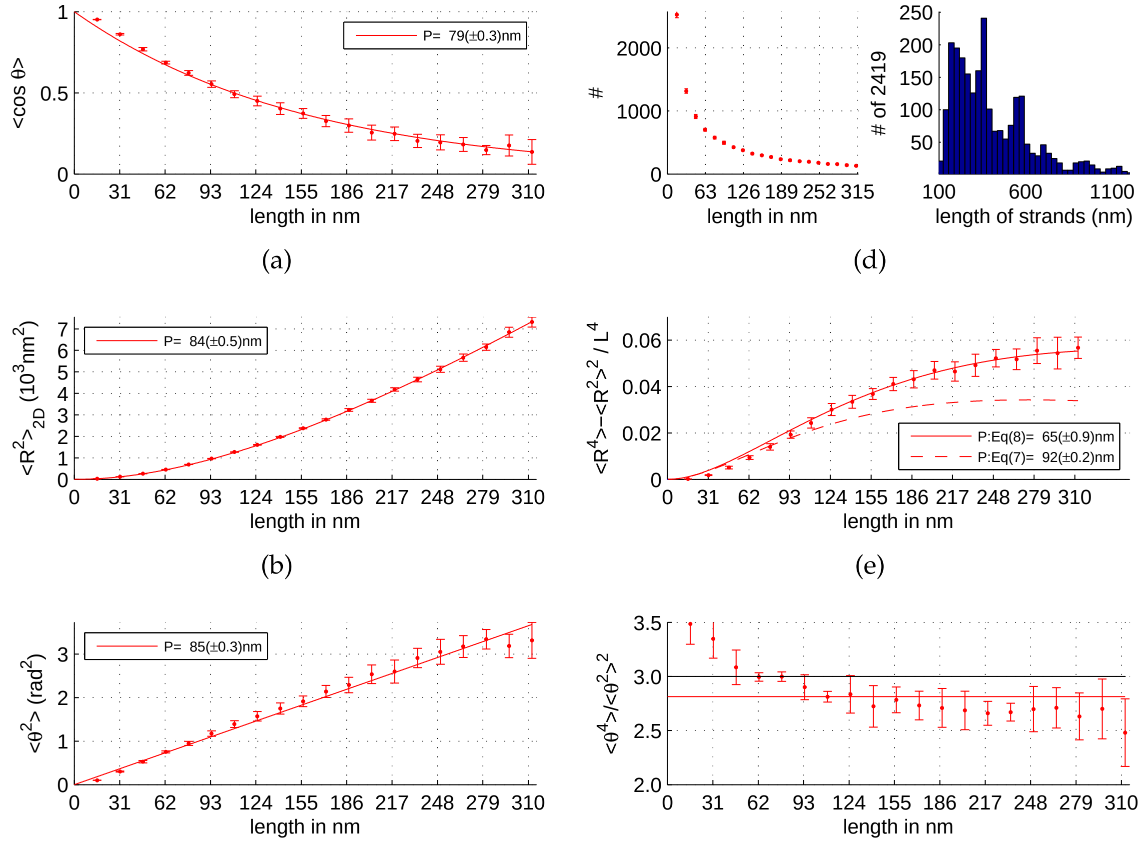
(a) (d)

(b) (e)

(c) (f)

Figure 9.2: Example of the data analysis for the 474 bp DNA fragment mounted onto polylysine film from 1 mM NaCl and imaged by EM. a)-c) show estimates of the persistence length based on $\langle\cos\theta(L)\rangle$ Eq.(9.5), $\langle R^2(L)\rangle$ Eq.(9.4) and $\langle\theta^2(L)\rangle$ Eq.(9.6) respectively; d) shows the number of unique used DNA strand elements to compute the different descriptors per contour length L and the length distribution of the analyzed molecules; e) shows the normalized difference $(\langle R^4\rangle - \langle R^2\rangle^2)/L^4$ Eq.(9.9) with fits for the 2D Eq.(9.8) and 3D Eq.(9.7) case; f) shows the kurtosis $\langle\theta^4\rangle / \langle\theta^2\rangle^2$ (which is equal to 3 for θ normally distributed around zero).

does not vanish. Not even for a length up to 3000 bp and $P_{3D} \approx 50$ bp, after which $P_{3D} = \sqrt{3/2}P_{2D}$ (for P_{3D}=100 bp the difference can be detected up to ~5000 bp). Therefore, the quantity $\langle D^4\rangle$ provides reliable means for determining the dimensionality of the deposited molecules and hence the persistence length by selecting the correct model. Methodologically, it means that we fit two sets of formulae (9.1-9.3) and (9.7) respectively (9.4-9.6) and (9.8) to the quantities measured from the experimental data. The test of Eq. (9.9) is used to make a proper choice for the dimensionality of the molecules and as a result determines the persistence length in 3D by selecting the corresponding set of formulae. The plots of $\langle D^4\rangle$ for 2D and 3D cases are depicted in Fig. 9.2(e). It is clearly seen that (i) the data can be described using the corresponding formula for 2D (Eq. 9.8) and (ii) the persistence length is very close to those calculated using formulae (9.4-9.6). In contrast, using formula (9.7) we failed to fit the data for all distance separations up to 120 nm. The depicted curve is the closest approximation to the experimental data with P_{3D} =35 nm. The DNA



(a) (d)

(b) (e)

(c) (f)

Figure 9.3: Example of the data analysis for 2981bp DNA with 10 mM KCl, 5 mM MgAc$_2$ mounted on mica and imaged with AFM; a)-c) show estimates of the persistence length based on $\langle\cos\theta(L)\rangle$ Eq.(9.5), $\langle R^2(L)\rangle$ Eq.(9.4) and $\langle\theta^2(L)\rangle$ Eq.(9.6) respectively; d) shows the number of unique used DNA strand elements to compute the different descriptors per contour length L and the length distribution of the analyzed molecules; e) shows the normalized difference $(\langle R^4\rangle - \langle R^2\rangle^2)/L^4$ Eq.(9.9) with fits for the 2D Eq.(9.8) and 3D Eq.(9.7) case; f) shows the kurtosis $\langle\theta^4\rangle / \langle\theta^2\rangle^2$ (which is equal to 3 for θ Gaussian distributed around zero).

molecules analyzed in Fig. 9.2 were 474 bp long and imaged by EM. Convincingly we found also for longer strands imaged by AFM in Fig. 9.3(e) that $\langle D^4\rangle$ best fits to the 2D formulae thereby supporting the conclusion that the DNA was imaged in a 2D equilibrium state.

Having validated the image processing, data analysis and the usage of the corresponding formulae (9.4-9.6) and (9.8) we calculated persistence lengths of DNA fragments ranging from 474 bp up to ~3000 bp for different salt conditions (NaCl 0-100 mM, KCl, MgAc$_2$), buffer composition (Tris, Hepes), nature of the imaging surface (glow-discharged carbon film, polylysine film) by EM and AFM. The results show (compare Table 9.1) that all four quantities used for the persistence length calculation give very similar results supporting our approach even further. It implies that regardless of deposition and imaging conditions DNA molecules behave as 2D polymers for lengths in the range ~ 10 - 120 nm (EM data) or ~ 10-300 nm (AFM data). EM data show that the persistence length of the molecules deposited from

1-100 mM NaCl is in the range 25-32 nm regardless of the nature of the deposition surface (glow discharged carbon film or polylysine film). At zero NaCl concentration we measured a smaller persistence length of 15-17 nm (not shown). Deposition from the 5 mM MgAc$_2$ and 10 mM KCl solutions led to a small but detectable decrease in persistence length (22-24 nm). In contrast, using the latter conditions for the deposition on a mica surface (AFM data) leads to a dramatic increase of the persistence length to 80-90 nm as calculated by using either quantity (Table 9.1).

Mounting conditions	Scored length [μm] (# molecules)	$P_{<R^2>}$ [nm]	$P_{<\cos>}$ [nm]	$P_{<\theta^2>}$ [nm]	$P_{<D^4>}$ [nm]	avg. [nm]
2.5mM NaCl$^{1)}$	$\sim$ 33 (210)	30	29	29	29	29
10mM NaCl$^{1)}$	$\sim$ 70 (451)	31	31	33	34	32
50mM NaCl$^{1)}$	$\sim$ 37 (235)	35	31	32	30	32
100mM NaCl$^{1)}$	$\sim$ 76 (490)	25	25	26	25	25
1 mM NaCl, polylisine$^{2)}$	$\sim$ 107 (685)	26	23	25	23	24
2.5 mM NaCl, polylisine$^{2)}$	$\sim$ 52 (332)	28	26	27	23	26
10 mM KCl, 5 mM MgAc$_2^{3)}$	$\sim$ 134	23	20	22	23	22
10 mM KCl, 5 mM MgAc$_2^{4)}$	$\sim$ 151	23	21	24	27	24
10 mM KCl, 5 mM MgAc$_2^{5)}$	$\sim$ 112	21	19	22	26	22
10 mM KCl, 5 mM MgAc$_2^{6)}$	$\sim$ 1000	89	80	81	66	79
10 mM KCl, 5 mM MgAc$_2^{7)}$	$\sim$ 563	101	91	91	76	90

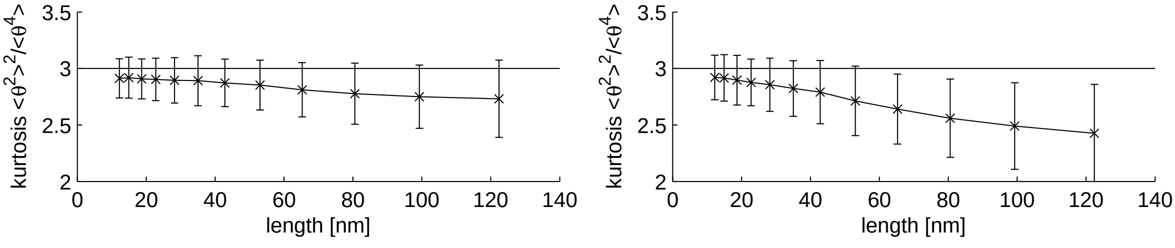
Table 9.1: Persistence length calculated using different statistical descriptors of the WLC model for various mounting and imaging conditions. 1) 474 bp DNA fragment, deposition onto glow discharged carbon film, EM data; 2) 474 bp fragment, deposition onto polylysine film, EM data; 3) mixture of 1144bp and 1837 bp fragments, deposition onto glow discharged carbon film, EM data; 4) 2981bp fragment, deposition onto glow discharged carbon film, EM data; 5) mixture of 474 bp and 2507 bp fragments, deposition onto glow discharged carbon film, EM data; 6) 2981bp fragment, deposition onto mica, AFM data; 7) mixture of 474 bp and 2507 bp fragments, deposition onto mica, AFM data.

9.2.3 Kurtosis and static bending of DNA

The homopolymer model describes the DNA flexibility by assuming an harmonic potential in the bending angle θ. This model results in $\langle\theta\rangle = 0$. The kurtosis is defined as

$$k = \frac{\langle\theta^4\rangle}{\langle\theta^2\rangle^2} \tag{9.10}$$

and is always equal to 3 regardless of the dimensionality of the molecules in this model. We found that for AFM data the kurtosis is close to 3 for distances $\sim\ 50-80$ nm along the DNA; for longer distances it remains within the range 2.7-2.8 (see Fig. 9.3(f)). However, for EM data the kurtosis was always below 3 showing a decrease to $\sim$ 2.5 for distances larger than 25 nm along the DNA (see Fig. 9.2(f)). For a homopolymer model, a deviation of the kurtosis from the expected value may be



(a)

(b)

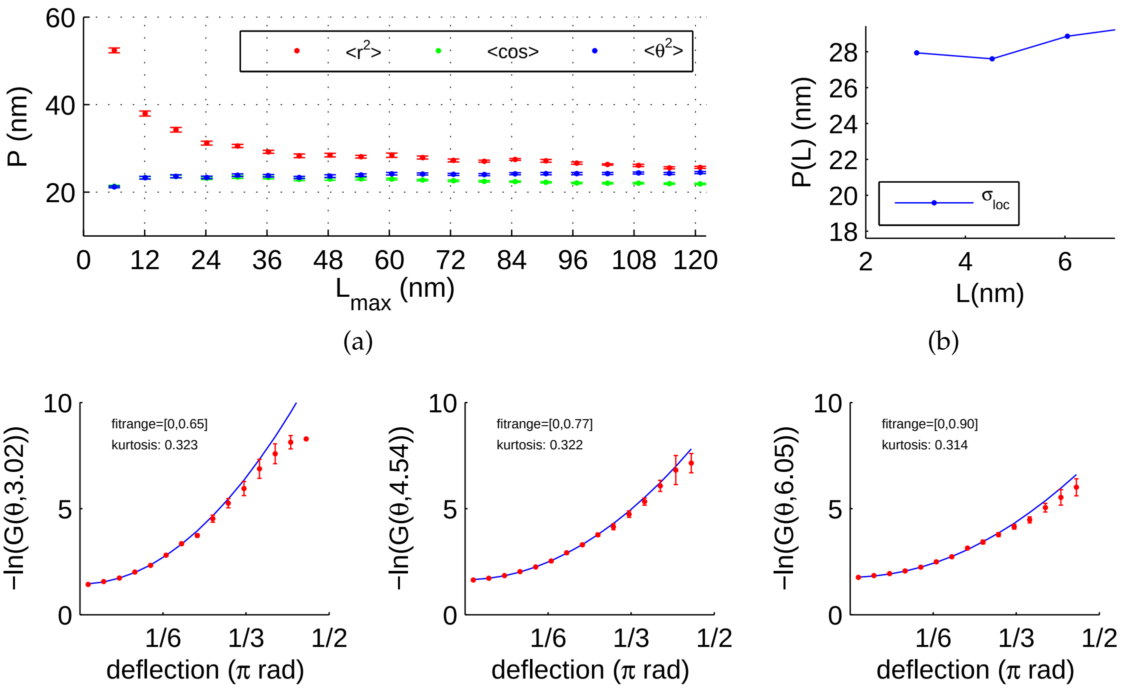
Figure 9.4: Simulated kurtosis for a non-zero static bending of $\theta_{stat} = 5.3°$ a) probability $p = 0.5$ and b) probability $p = 0.53$ of the sign. The simulation was performed on 500 molecules of 474bp chain length (see for details Materials and Methods) and averaged over 500 independent realizations. The error bars indicate 1σ spread.

indicative for either an inadequate procedure for image processing and/or analysis, or the presence of non-equilibrium processes during deposition. Another possibility implies that harmonic potential accounting for DNA bending is still quadratic but around a non-zero value (heteropolymer model). Having validated the image processing, the data analysis and the applicability of the formulae used, we consider the last possibility. Therefore we introduce a small, surface-induced static bending at each dinucleotide step. For the simple assumption that all static bends occur in plane with identical amplitude but random sign, we derived an expression for the kurtosis accounting for the average static bending angle and separation distance (see Online Supporting Material). The kurtosis for the heteropolymer model,

$$k = 3 - \frac{2}{n}\frac{\theta_{stat}^4}{(\theta_{dyn}^2 + \theta_{stat}^2)^2} . \tag{9.11}$$

shows that the kurtosis rapidly approaches 3 as the length along the DNA n increases. In this case the sign of the static bends was chosen randomly with equal probabilities. To calculate the average amplitude of the static bends we recall that $P \approx \frac{2}{\langle\theta_{dyn}^2\rangle + \langle\theta_{stat}^2\rangle}$ where $\langle\theta_{dyn}^2\rangle$ and $\langle\theta_{stat}^2\rangle$ stand for the dynamic behavior of polymer chain and static bending, respectively [134, 124]. Using an identical procedure for DNA deposition as we used for EM analysis, Cognet et al. [20] found that for DNA molecules confined to a plane $\theta_{dyn} \approx \sqrt{\langle\theta_{dyn}^2\rangle} = 4.6°/\text{bp}$. Since we are looking for angle displacement in 2D we have to use the corresponding value for the persistence length. Our EM measurements show that P_{3D} is in the range of 15-32 nm (see Table 9.1) what corresponds to $P_{2D} \sim$ 90-190 bp. It means that $\theta_{stat} \approx \sqrt{\langle\theta_{stat}^2\rangle}$ is in the range $3.7° - 7.2°$.

For an intermediate value of $\theta_{stat} = 5.3°$ we performed Monte-Carlo simulations of polymer chains and evaluated the expected kurtosis for different lengths along the DNA n and different probabilities p of positive and negative sign of θ^0. Given the assumption of $p = 0.5$ and a large number of molecules ($> 10^4$), the kurtosis is indeed exactly 3 for any n. If we decrease the number of molecules to similar numbers as encountered in the experiments (~ 500) we find a decrease of the kurtosis with increasing n for any $\theta_{stat} \neq 0$ (see Fig. 9.4) for any value of p. Higher values of p yield smaller kurtosis for larger distances along the DNA similar to the experimental



(a) (b)

(c)

Figure 9.5: a) Fitted persistence length based on $\langle R^2 \rangle$, $\langle \theta^2 \rangle$ and $\langle \cos\theta \rangle$ as a function of the distance length L along the DNA for the same molecules as in Fig. 9.2, b) fitted persistence length (blue) for very short distances only based on the histograms shown in c); c) negative logarithm $-\ln G$ Eq.(9.12) of occurrence of the deflection angle θ for very short segments [L=3.02, 4.54, 6.05 nm]. The deflection angle histogram is computed from a few thousand bends.

data, EM data in particular. Considering the fact that we only have a few hundred molecules no conclusion can be drawn about the exact value of p. However, it must be in the range 0.5 ± 0.03 given the simulations. In control simulations with $\theta_{stat} = 0°$ the kurtosis was always 3 for any number of molecules.

Therefore, it becomes evident that the experimental data can be explained by a modified homopolymer model accounting for static bending together with limited sampling. However, in contrast to AFM data, EM data require that static bending should be larger mostly due to surface-induced bending in addition to the intrinsic dynamic and static bendings.

9.2.4 DNA bending at short distances

We investigated flexibility or bending of DNA over very short distances as done by Wiggins et al. [158] in an AFM study. Their measurements imply that the elastic energy of highly bent DNA conformations is lower than predicted by the WLC model. That is, on short length scales, large angle bends occur more frequently than predicted by the WLC. In Fig. 9.5(c) we computed the negative logarithm of the observed bend angle probability distribution function $G(\theta; L)$ for short distances L for the molecules investigated in Fig. 9.2. In Eq.(9.12) we have derived a formula (see Online Supporting Material)

$$G(\theta, L) = \mathrm{erf}\left(\sqrt{\frac{P}{2L}}\left(\theta + \frac{\Delta\theta}{2}\right)\right) - \mathrm{erf}\left(\sqrt{\frac{P}{2L}}\left(\theta - \frac{\Delta\theta}{2}\right)\right) \qquad (9.12)$$

that takes binning effects in the probability density function of the bending angle θ into account. This enables exact fitting of the persistence length also from the histogram. The fitted persistence length is depicted in Fig. 9.5(b) in blue. For three distances along the DNA $L = 3.02, 4.54$ and 6.05 nm we found $P \sim 28$ nm. We did not find any deviation from a Gaussian distribution of the bend angles over any distance, and in particular not over short distances (see Fig. 9.5(c)). The data shown here are typical for all investigated supports and mounting conditions used for EM. For AFM data we could not investigate the bending on such short length scales due to a lack of resolution.

9.3 Discussion

Microscopy techniques permit the analysis of conformations of long flexible polymer molecules, DNA in particular, in a dried state. The analysis requires that (i) molecules irreversibly adhere to an imaging surface and (ii) all segments along the length should be imaged in an identical manner though possibly with some intensity modulations reflecting the processes of molecule deposition and/or detection. However, these requirements do not ensure that the conformation of deposited molecules will be unique (2D or 3D) and identical; moreover, molecules may be of mixed conformations. In addition, the apparent conformation of the molecules may depend on their length. The conformational state of deposited molecules is traditionally analyzed by fitting experimental datasets with the formulae accounting for statistical properties of equilibrated molecules either in 2D or in 3D. Goodness of fit, the intrinsic conformity of the calculated values for the persistent length together with agreement with literature data is often considered as a strong support for the techniques used for molecules deposition and image analysis. In contrast, inconsistency of the experimental data with theoretical expectations calculated using one of the descriptors, e.g. $< R^2 >$, is often interpreted as a result of non-equilibrium processes prior to adhesion of the molecules, e.g. "trapping" of the molecules by the deposition surface [112]. Goodness of fit, however, only shows the validity of the model used without guarantee that the persistence length will be close to $\sim$ 50 nm. Deviation of the measured persistence length (using, e.g. $\langle R^2 \rangle$ descriptor) from the expected value of $\sim$ 50 nm, in some instances, allowed authors to speculate about the projection of 3D equilibrium DNA conformation onto 2D imaging plane [65].

Our data show that DNA molecules of $\sim$500 - 3000 bp long deposited onto a solid support (either glow-discharged carbon film or polylysine film or mica) behave as an equilibrated 2D polymer chain for distances along the DNA up to $\sim$120 nm (EM measurements, Fig. 9.2(e)) or 300 nm (AFM measurements, Fig. 9.4(e)), though of different persistence length. The overall procedure used for EM imaging (deposition onto glow-discharged carbon film together with uranyl staining) had been validated in a sense that it does not change DNA dynamic behavior, i.e. $< \theta^2_{dyn} >$ is close to 4.6° [20]. However, it does modulate the apparent persistence length. We believe that these modulations of persistence length (decreasing) are mainly due to distortions of the DNA backbone. These distortions do not impair the harmonic potential of DNA flexibility at each dinucleotide step since the conformation of the molecules along long distances follow theoretical predictions with high accuracy. We rather believe that adhesion to a charged (imaging) surface induces an additional static bending. We note that intrinsic static bends, introduced to describe sequence-dependent DNA deformations, i.e. curvature (reviewed in [124]), can account for $\sim 3^\circ$ [54] on average for the molecules confined to a plane. Our data show that the static bending should be in the range $3.7^\circ - 7.2^\circ$ (in plane), most probably close to 5° implying that deposition onto carbon film or polylysine film induces additional DNA bending (kinking). In this line, AFM data showing the kurtosis in the range 2.7-2.8 are in agreement with the heteropolymer model of DNA exhibiting intrinsic static bending that is supported by our Monte-Carlo simulations (see Fig. 9.4).

In support of our conjecture of equilibrated 2D polymers we stress that conformity of DNA conformation to the theoretical expectations was found for distances along the DNA beyond 100-150 nm, a value after which a significant deviation of DNA conformation from the WLC model is usually found [63, 138, 93, 101]. This deviation is often interpreted as a 3D state of imaged DNA due surface "trapping" [63, 138, 93]. Fig. 9.3 shows that under our deposition/preparation conditions, DNA conformation can be well fitted by any descriptor used here till separation distances $\sim$ 300 nm. Attempts to fit $\langle R^2 \rangle$ using scaling formulae for polymer wandering in 2D or 3D as earlier suggested [63, 138] did not produce any physically relevant result. Importantly, at all separation distances DNA behaves in accordance with the WLC model for either combination of mounting/imaging conditions.

The whole analysis presented here is based on fitting the known homopolymer formulae (9.4-9.6 and 9.8) to the experimental data. However, as we argued in the case of EM mounting, an additional (induced) static bending is required; we have to justify usage of these formulae for heteropolymer molecules. Using Monte-Carlo simulation we generated a set of 474 bp long molecules using the harmonic potential with different values of $< \theta_{dyn} >$ (corresponding to P_{3D} ranging from $\sim$ 20 nm to $\sim$ 50 nm) and $\theta_{stat} = 5.3^\circ$, and analyzed thus simulated molecules as described. The results (see Fig. 9.10) clearly show that (i) fitting with formulae (9.4-9.6 and 9.8) give similar values for the apparent persistence using either descriptor and (ii) thus calculated apparent persistence length is in a good agreement with the effective persistence length determined via Eq. 9.31. The kurtosis decreases from 3 to $\sim$ 2.5 as the length along the DNA increases up to $\sim$ 100 nm in agreement with EM data. Attempts to fit the data with formula 9.7 corresponding to the 3D conformation of the molecules, failed for distances beyond 40 nm implying their 2D state. Importantly, fitting the model to short distances (4 nm and 6 nm) using Eq. 9.12 led to a persistence length similar to those calculated using Eq. 9.4-9.6 and 9.8. Together, these results validate the usage of the formulae 9.4-9.6 and 9.8 for the purpose of analysis of heteropolymer molecules.

It is known that in general persistence length of DNA decreases as the concentration of monovalent cations increases in solution [57, 83, 5, 80] reaching a value close to 50 nm. Similar behavior is observed for multivalent cations though the limiting value for the persistence length may be somehow smaller [57, 83, 5, 80]. Deposition onto negatively charged surface, e.g. freshly cleaved mica, mediated by the divalent cations, e.g. Mg^{2+}, may preserve persistence length close to 50 nm [112, 101]. At some instances higher values ($\sim$ 80-140 nm) were reported [112, 93, 37, 162] using $< R^2 >$ or $< \theta^2 >$ descriptors. In the latter case a clear deviation from the expected strait line was found [112] for separations $>$ 100 nm indicative for the contribution of excluded volume effects for molecules with contour lengths larger than 20 persistence lengths. In our case, AFM data show that either descriptor can be fit by the analytical formulas giving consisting values for the persistence length. Since the goodness of fit was high for DNA separations in the range $\sim$ 15-300 nm (see Fig. 9.3), we infer that apparent persistence length was similar regardless of separation along DNA contour implying small contribution of excluded volume effect. Thus, our data are in accord with the results of Frontali et. al. [48, 47] showing absence of excluded volume effect for cytochrome c adsorbed DNA with contour lengths up to 30 persistence lengths.

Large values for the persistence length can be explained by modulation of pre-equilibrated DNA molecules close to the mica surface during rinsing step. Rinsing of the sample and its drying, steps that are required for the fixation of DNA onto surface and imaging in air, may lead to the modulation of apparent DNA conformation as described [19] in a way depending on the forces occurred between pre-equilibrated molecules and mica surface. Since at many instances rinsing buffer is a pure water, the Debye length increases leading to an increase of persistence length [48, 57, 83, 5]. Deposition from low salt buffer onto positive charged surfaces may lead to a significant decreasing apparent persistence length from $\sim$ 40 nm to $\sim$ 11 nm in a manner depending on surface charge density [101]. The authors proposed that surface charge density modulates a fractional neutralization of DNA phosphate groups leading to a significant increase in DNA flexibility. We note that the glow-discharged carbon film and polylysine film used here are positive charged surfaces. Therefore, we cannot rule out this mechanism accounting for the decreased persistence length of DNA molecules from EM experiments, though we believe that induced DNA static bending (kinking) is a more favorable one in a view of notable modulation of the kurtosis.

Recent studies by Wiggins et al. [158] and Mathew-Fenn et al. [85] argued that the classical model of DNA duplex as an ideal elastic rod fails to describe either increased DNA flexibility [158] or end-to-end distance distribution [85] on short length scales of 5-10 nm. For instance Wiggins et al. [158] observed a non-Gaussian potential for the bend angle on these length scales. In our EM data we had the resolution to test this finding. However, we do not see any deviation from a Gaussian distribution of the bend angle down to 3 nm, i.e. we do not find more high angles in the histograms of the negative logarithm $-\ln G(\theta; L)$. Our estimation of the persistence length P as a function of distance L on experimental data is consistent from $-\ln G(\theta; L)$ Eq.(9.12) and $P(L)$, compare Fig. 9.5(b)). The fact that we do not observe an increased flexibility could be due to a lower persistence length of $\sim$ 20 nm in comparison to $\sim$ 50 nm in their study. A lower persistence length (higher flexibility) makes it less probably to observe deviations at the same separation distance as in [158]. To observe angle fluctuation with the same probability we would need to evaluate at the same P/L ratio, i.e. at separations $L < 2$ nm. This is beyond the imaging resolution and the subsequent image analysis. Note that our Monte-Carlo simulations of the three short length scales used show consistent estimation of P down to separations of $L = 4$ nm for similar noise levels as observed in the EM images (see Fig. 9.8(d)). This rules out that the image processing algorithm is "too stiff" to follow the DNA backbone exactly. We always find more higher angles than Gaussian distributed (see Fig. 9.5(c)), but with so little occurrence that no significant conclusions can be drawn. Together, our findings do neither confirm nor contradict these studies.

Lastly, we note that the analysis of specific DNA-protein complexes shows that in many circumstances the DNA backbone may be significantly distorted (relative to the solution structure) forming a bend or kink upon formation of specific DNA-protein complex [28]. Recent data exploiting molecular dynamic simulations indicate that DNA flexibility and deformability (so called "indirect readout") play a key role in discrimination specific sequences and stabilization of the complex structure upon interaction with proteins [98, 96]. We therefore speculate that deformations of DNA backbone detected as induced bending/kinking upon its interaction with highly charged surface is in agreement with these data. Moreover, this behavior of DNA is an intrinsic property, the extent of which possibly modulated by the strength of DNA -surface interactions.

9.4 Methods and Materials

Sample preparation

Supercoiled pPGM1 plasmid DNA (2981 bp) was purified using Qiagen kits (Hilden, Germany) [19]. Linear DNA fragments were obtained after digestion with either EcoRI (2981 bp fragment), EcoRI/ ScaI (1444 bp and 1837 bp fragment) or PvuII (474 bp and 2507 bp fragments) and purified using gel filtration chromatography on a Superose 6 column (SMART system, Amersham Biosciences). A 474 bp fragment was obtained by additional purification through an anion exchange Waters Gen-Pak FAX column. All samples were stored in a buffer of 10 mM Tris-HCl, pH 7.5, 10 mM NaCl, 0.1 mM Na_3EDTA at a concentration of 200 - 400 μg/ml.

Electron & Atomic force microscopy

Two different procedures were used for the deposition of DNA molecules for electron microscopy [19]: adsorption to glow-discharged carbon film and to polylysine film. The samples were analyzed with a Philips CM12 electron microscope in a dark-field mode at a magnification of 28,000-35,000. The sampling density of the digital images was 1.21 nm/pixel or 1.51 nm/px.

For AFM DNA was deposited onto the surface of freshly cleaved mica (Muscovite, Plano GmbH, Germany). Samples were a drop of DNA solution (10 μl) at a concentration of 2 μg/ml from buffer (B) was placed onto the surface of freshly cleaved mica (Muscovite, Plano GmbH, Germany). Samples were scanned using a Digital Instruments MultiMode scanning probe microscope Nanoscope IIIa (Veeco) operating in Tapping Mode. The sampling density of the digital images was 7.81 nm/pixel. For details see Online Supporting Material.

Image & Data analysis

The image analysis software was custom developed using in the Matlab toolbox DIPimage. The software is provided as Matlab (The Mathworks, USA) scripts which can be freely downloaded from
www.diplib.org/home22266. Shortly the algorithm uses an improved Fast Marching algorithm [59] after identification of the relevant molecules to find the DNA centerline. The image analysis algorithm was extensively validated on images generated by Monte Carlo simulations. In the data analysis extra precaution was taken in sampling the strands to avoid correlation. Reusing all data for computing the characteristic quantities for each length along the DNA yield highly correlated points. To avoid this, we divided each DNA strand into length segments randomly drawn from a predefined set of lengths such that no piece of strand is used twice. For details see Online Supporting Material.

Acknowledgments

We thank G. Heim for help with the image acquisition and T.M. Jovin for discussion. F.G.A. Faas was partially supported by Rolling Grants program 94RG12 of the Netherlands Organization for Fundamental Research of Matter (FOM). DIC was awarded by a Wellcome Trust VIP grant.

Supplementary Material

9.5 Sample preparation

Supercoiled pPGM1 plasmid DNA (2981 bp) was purified using Qiagen kits (Hilden, Germany) as described in [19]. Linear DNA fragments were obtained after digestion with either EcoRI (2981 bp fragment), EcoRI/ ScaI (1444 bp and 1837 bp fragment) or PvuII (474 bp and 2507 bp fragments) and purified using gel filtration chromatography on a Superose 6 column (SMART system, Amersham Biosciences). A 474 bp fragment was obtained by additional purification through an anion exchange Waters Gen-Pak FAX column. All samples were stored in a buffer of 10 mM Tris-HCl, pH 7.5, 10 mM NaCl, 0.1 mM Na_3EDTA at a concentration of 200 - 400 μg/ml.

9.5.1 Electron microscopy

Two different procedures were used for the deposition of DNA molecules: adsorption to glow-discharged carbon film and to polylysine film [19]. In the first procedure, a stock solution of DNA was 100-200 fold diluted either in buffer (A) containing 1 mM Tris-HCl, pH 7.5, 0-100 mM NaCl or in buffer (B) containing 5 mM Hepes, pH 7.5, 5 mM $MgAc_2$, 10 mM KCl. The final DNA concentration was 0.2-1 μg/ml. A drop of this solution (6-8 μl) was placed onto the surface of carbon film mounted on an EM grid. Carbon films, 3-4 nm thick, were glow-discharged in the presence of pentylamine vapor (residual pressure 150 mTorr, discharge current 2-3 mA, duration of discharge 30 seconds) as described elsewhere [19]. The adsorption was continued for one to two minutes, then the grids were rinsed with a few drops of 2% (w/v) aqueous uranyl acetate, blotted with filter paper and air-dried.

In the second procedure, adsorption to polylysine film was carried out as described in [19]. In summary, carbon-coated EM grids were glow discharged in air (residual pressure (200 mTorr, discharge current 8-9 mA, duration of discharge 30 seconds, Bal-Tec MED 020 coater) and immediately coated with poly-L-lysine (molecular mass 2000, Sigma) by adding 8 μl of its aqueous solution at a concentration 3 μg/ml for one minute. The grids were then drained with a long tip connected to a vacuum-connected aspirator and air dried. A drop of DNA solution (6-8 μl) buffer (A) was placed onto the polylysine film and DNA was allowed to adsorb for 1.5-2 minutes. The grid was rinsed with a few drops of 2% (w/v) aqueous uranyl acetate, blotted with filter paper and air-dried.

The samples were analyzed with a Philips CM12 electron microscope in a dark-field mode at a magnification of 28,000-35,000. The negatives were scanned with a DuoScan T2500 scanner (Agfa, Germany) at 600 - 1200 dpi. The sampling density of the digital images was 1.21 nm/pixel or 1.51 nm/px (See Fig. 9.6 and 9.1). The absolute value for DNA rise was found to be 0.32 - 0.33 nm. In Fig. 9.6 we show a panel of images of various DNA fragments mounted under different conditions. For printing, images were flattened using a high-pass filter with a radius of 250 pixels and subsequently adjusted for contrast/brightness with inverted grayscale using Adobe Photoshop^{TM}.

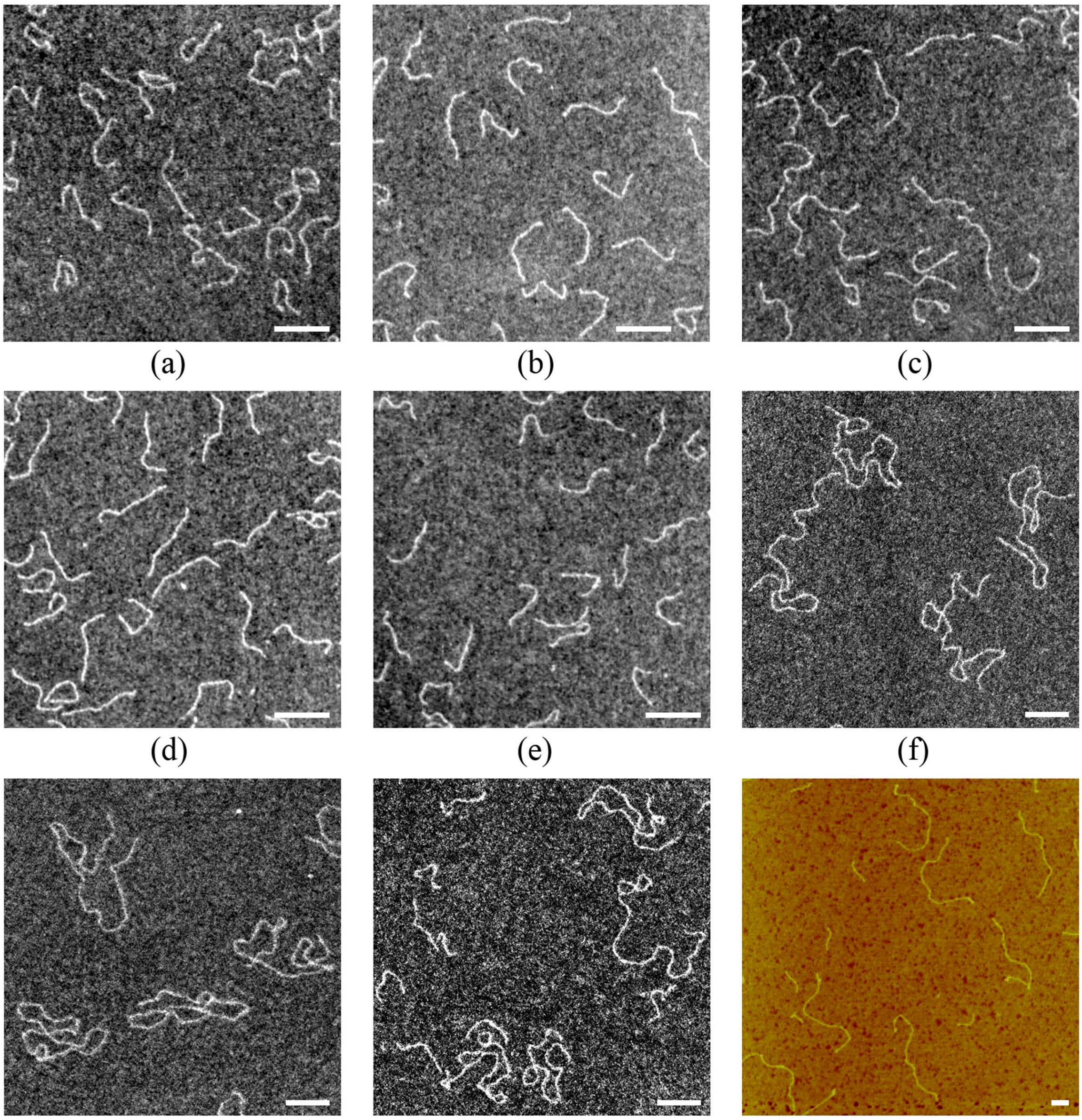
9.5.2 Atomic force microscopy

For AFM imaging samples were prepared as described [19]. Briefly, a drop of DNA solution (10 μl) at a concentration of 2 μg/ml from buffer (B) was placed onto the surface of freshly cleaved mica (Muscovite, Plano GmbH, Germany) for two minutes, then rinsed with 1 ml of water, blotted with filter paper; the remaining water was blown away by the flow of compressed air. Samples were scanned using a Digital Instruments MultiMode scanning probe microscope Nanoscope IIIa (Veeco) operating in Tapping Mode. Scan rates varied from 3 to 5 Hz. Commercial silicon probes TESP-100 (Veeco) with a typical resonant frequency of around 300 kHz or ultrasharp NSC15 cantilevers (Mikromash, USA) were used throughout the experiments. The sampling density of the digital images was 7.81 nm/pixel (See Fig. 9.6 and 9.1). The absolute value for DNA rise was 0.31 nm in accordance with previously published data [112].

9.6 Image analysis

The image analysis software was custom developed using in the Matlab toolbox DIPimage [81]. The software is provided as Matlab (The Mathworks, USA) scripts which can be freely downloaded from `www.diplib.org/home22266`. It provides the possibility to analyze images obtained from either EM or AFM imaging given sufficient image quality. A brief user manual and a few test images together with presegmented DNA strands are provided to make the software easily accessible to the field of experimental polymer studies on flat support.

Although the detection and tracing of the molecules was done fully automated, a human supervisor could reject erroneously segmented or traced molecules in an interactive step. After automated detection of DNA strands, a path through its centerline is extracted by an improved Fast Marching algorithm [59] resulting in x, y coordinate pairs tracing the molecule. They are resampled such that consecutive pairs are separated by 1 nm along the centerline of the molecule. The actual processing is divided into i) background subtraction, ii) coarse segmentation of the DNA molecules, iii) end point refinement and iv) tracing the centerline of the molecules.
First, a gradually changing background is removed by the subtraction of a low pass filtered version of the original image with a large filter kernel (Gaussian filter with $\sigma = 25$ nm). This is equivalent to high-pass filtering and results in a background corrected image I. See Fig. 9.1(a)) for an example before pre-processing.
The segmentation of the DNA strands is difficult due to the grainy structure caused by uranyl staining in combination with the high-resolution of the EM. As a result, the contrast along the molecules is far from constant and resembles a string of beads. To reduce the intensity variation along the contour the images are smoothed in by coherency enhancing diffusion step [154], resulting in I_{CED}. A single threshold is still not sufficient, therefore we use an anchor skeleton [150] to segment the molecules, i.e. a thinning operation in which the "anchors" are not allowed to be removed. The anchors are obtained by a relative low threshold, i.e. $I_{anchor} = I_{CED} < t_{low}$ and the image to be thinned by $I_{high} = I_{CED} < t_{high}$ with $t_{low} < t_{high}$. From this skeleton all branches, bifurcations and loops are removed such that the end points remain. In the following we obtain a better estimate of the end-points of the strand. First a region



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 9.6: Examples of EM (a-h) and AFM (i) images of DNA fragments deposited under various conditions and used for measurements. a) 474 bp fragment, 2.5 mM NaCl, carbon film, b) 474 bp fragment, 10 mM NaCl, carbon film, c) 474 bp fragment, 50 mM NaCl, carbon film, d) 474 bp fragment, 100 mM NaCl, carbon film, e) 474 bp fragment, 2.5 mM NaCl, polylysine, f) 2981 bp fragment, 50 mM NaCl, carbon, g) 2981 bp fragment, 5 mM MgAc2, 10 mM KCl, carbon, h) mixture of 474 bp and 2507 bp fragments, 5 mM MgAc2, 10 mM KCl, carbon, i) mixture of 474 bp and 2507 bp fragments, 5 mM MgAc2, 10 mM KCl, mica. Scale bars, 100 nm.

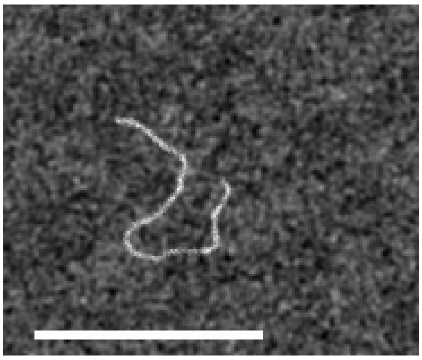


Figure 9.7: Monte-Carlo simulation of a 474 bp long DNA homopolymer with $P = 55$ bp (18.7 nm) and added Gaussian and Poisson noise to match visually the experimental data. Scale bar, 100 nm.

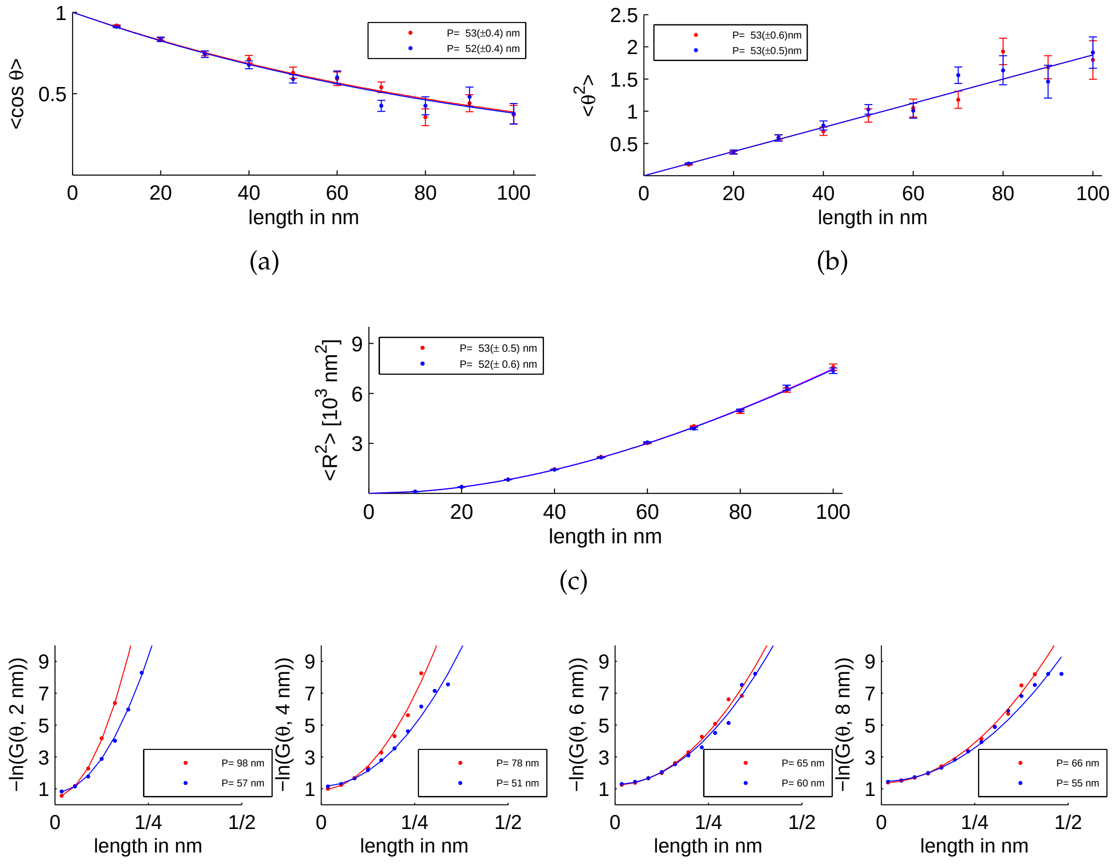
growing algorithm is applied to the pruned skeleton, i.e. the skeleton is allowed to grow into regions for which $I < t_{edge}$, where t_{edge} is halfway between fore- and background value of the flattened image I. From this generated mask we determine the center point by a skeleton operation which removes loose ends. The point in the mask with the largest distance, measured through the mask, from the mask center is the first end point E_1. The second end point E_2 is given by the point in the mask with the largest distance to E_1. Now a new anchor skeleton is made of the mask with E_1 and E_2 as the only anchors and the outcome is pruned. This skeleton is dilated to serve as the final mask for the tracking algorithm. In cases where the dilation would merge two regions or parts of one strand it is terminated at those specific positions. The centerline of the DNA molecules is found using an improved Fast Marching algorithm [59]. The centerline is the minimum-cost path between the two end points E_1 and E_2. The minimum arrival time T along all possible paths P connecting the two points within a single mask is given by

$$T = \min_{\forall P_{E_1 \to E_2}} \int I_{cost}(P(s))\, ds \tag{9.13}$$

where I_{cost} is the cost map which is given by the inverse of the local travel speed. Here the cost map is derived directly from the flattened images by $I_{cost} = (\frac{I - min(I)}{max(I) - min(I)})^\alpha + 1$, with $\alpha = 3$. The minimum cost path from E_1 to E_2 is found by solving the Eikonal equation

$$|\nabla U| = I_{cost} \tag{9.14}$$

with U the arrival time map and initial condition $U(E_1) = 0$. We use a fast marching algorithm [3] for solving Eq.(9.14). From the arrival time map the estimated centerline can be extracted by descending along the opposite gradient direction starting at point E_2. Due to the smoothness of the integrated cost images one can obtain subpixel accuracy in the location of the minimum-cost path. At this point we have a sequence of points P (at spacing 1/3 of a pixel) tracing the DNA centerline from E_1 to E_2. Normal fast marching algorithms will always result in a path that is shorter and stiffer than the centerline of the underlying one-dimensional structure unless the structure is straight. To eliminate this problem the iterative scheme as presented in [59] is used. This scheme basically deforms the image space such that in the deformed space the underlying structure becomes straight (see Fig. 9.1(c) where red



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

Figure 9.8: Result for 500 Monte-Carlo simulated images of homopolymer 474 bp DNA with $P = 159$ bp (52.7 nm) with (red) and without noise (blue). Only non self-intersecting molecules are processed; a-c) show estimates of the persistence length based on $\langle\cos\theta(L)\rangle$ Eq. 9.5 and $\left\langle\theta^2(L)\right\rangle$ Eq. 9.6 and $\left\langle R^2(L)\right\rangle$ Eq. 9.4 respectively; d) shows the negative logarithm $-\ln G$ Eq. 9.17 of the occurrence of the deflection angle θ for $L = 2, 4, 6$ and 8 nm.

indicates the initial path and blue the path after 25 iterations upon convergence). Finally, the endpoints are refined from the anchor skeleton with subpixel accuracy by extending the strand along the tangent of the path and finding the closest point with the value t_{edge}. Fitting a b-spline through this new path gives a smooth piecewise polynomial description of the DNA strand which is used in the actual data analysis. Note that the end parts, i.e. the first/last 3 pixels, of the estimated centerline are only used for the determination of the length of the DNA strands, for all other measurements the data from these parts is ignored.

9.7 Image analysis validation

The image analysis algorithm was validated on images generated by Monte Carlo simulations of the 2D WLC, i.e a homopolymer model with angle distribution according to Eq.(9.17). The evaluation was performed on noise free images and on images with added Gaussian and (correlated) Poisson noise to match visually the

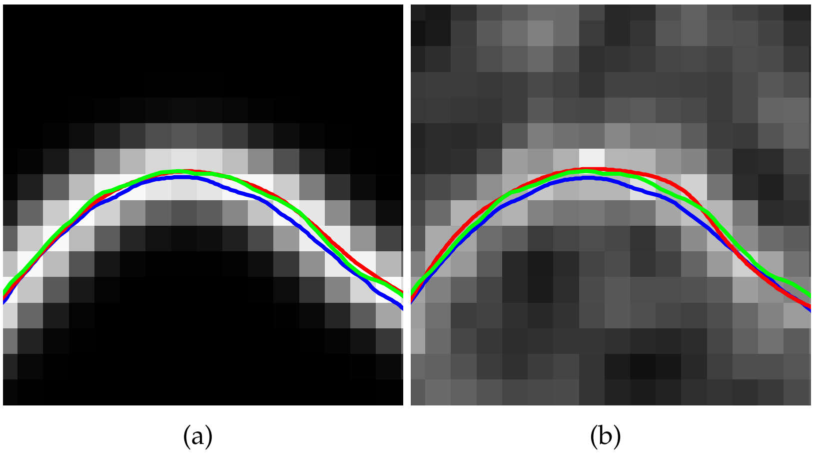

(a) (b)

Figure 9.9: Part of an image generated with Monte-Carlo simulation of 474 bp DNA with $P = 55$ bp (18.7 nm). The red line connects the simulated coordinates spaced at one bp of the DNA strand. In blue is depicted the initial found centerline and in green the finally used centerline; a) for the noise free case, b) for added Gaussian and Poisson noise to visually match the experimental data. Scale: 1 pixel corresponds to 1 nm.

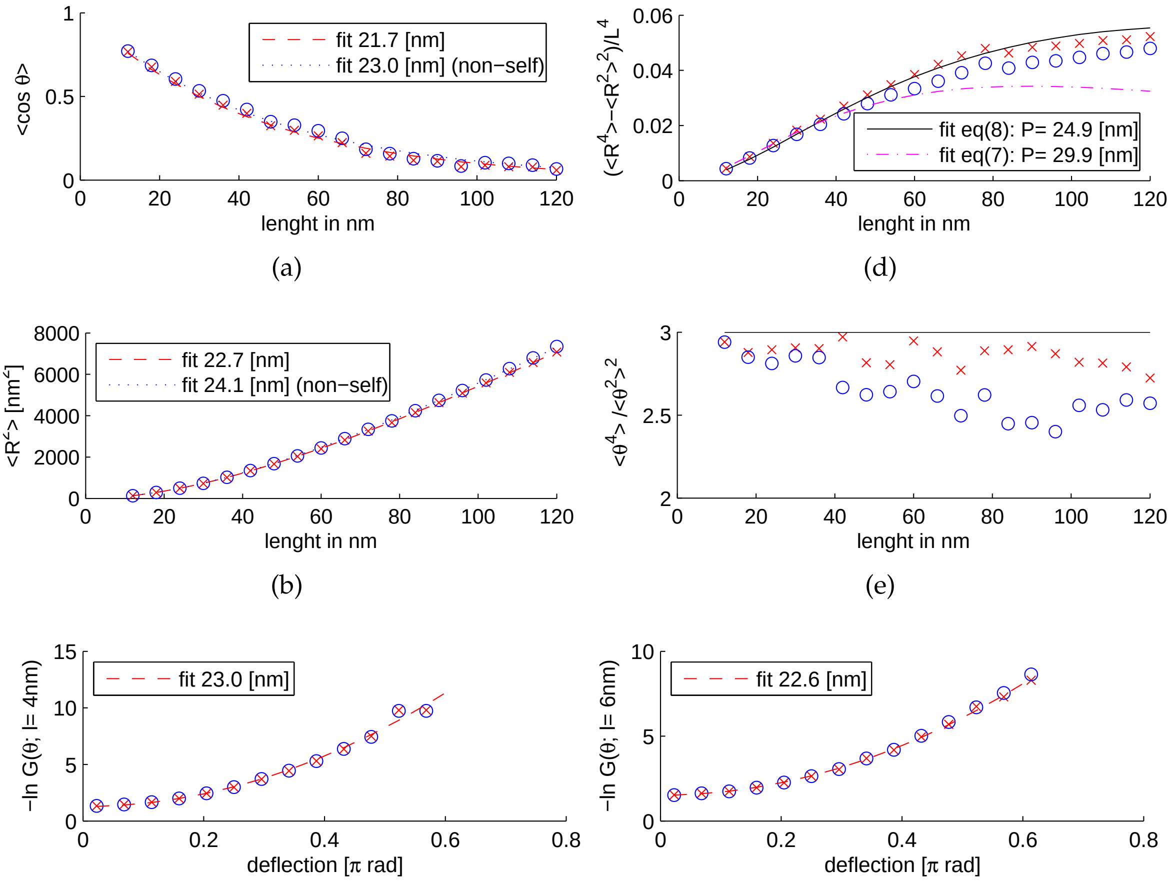
experimental data (see Fig. 9.7). A DNA strand was generated by adding Gaussian blobs of $\sigma = 1$ nm at the simulated locations to a test image at spacings of 1 bp (image sampling pitch 1 pixel = 1 nm). In Fig. 9.8 we depict the result of an analysis applied to 500 simulated images containing DNA molecules of 474 bp length with and without noise (only the non-self intersection molecules were processed).

The algorithm permitted estimation of the underlying persistence length, the kurtosis and the bend angle histogram. We validated the image processing method for polymers with persistence lengths ranging from 18.7 nm to 52.7 nm (55 bp to 159 bp). The bias in the found persistence length values was about $\sim$ 5%, if the fit was performed in the range $L \in [0, 2P]$. The bias was always positive, i.e. the persistence length was too high. The kurtosis was found to be constant with value 3. The bending potential $G(\theta, L)$ was verified for $L = 4, 6$ and 8 nm and also here the fitted persistence length was within 5% of the ground truth. Generally, the method performed better on molecules with larger persistence length (stiffer), e.g. for $P = 52.7$ nm the error was <2%. Given the spread in the estimate of the persistence length from the true simulated coordinates of $\sim$ 2% for 500 molecules and $\sim 4\%$ if only the non-self-intersecting molecules are considered, we conclude that the image processing retrieves the correct coordinates of the DNA centerline. In Fig. 9.9 we show the extracted centerlines for simulated images. The red line connects simulated DNA positions, the blue line is the initial centerline found by the fast marching and the green line shows the centerline by the iterative procedure [59]. The latter resembles the ground truth (red) very well in the noise free case and even for the noisy case the correspondence is good.

We observed that the apparent length distribution of the imaged 474 bp DNA as shown in Fig. 9.2(d)) is narrow. Traditionally, the length of all molecules found is set to the a-priori know length of the fragments for the analysis [112, 113]. However, the narrow distribution stimulated us to calibrate the magnification once for all images and keep the apparent length differences for the imaged ensemble of polymer. A comparison of both methods showed excellent agreement (found persistence length within 0.2 nm which equals the fitting error). Therefore we used a magnification calibration for all analysis. We emphasize that once the procedure is established, we can use segments of the molecules for the measurements instead of full-length. This greatly enhances the total scoreable length of DNA molecules in each data sets as overlapping molecules can be analyzed in parts.

9.8 Data analysis

Given a set of coordinates that trace the DNA backbone we compute the statistical quantities such as $\langle R^2 \rangle$ as a function of the contour length l. Extra precaution was taken in sampling the strands to avoid correlation between points in the graph. Reusing all data for computing the quantities for each length along the DNA yields highly correlated points, hence a smooth curve. To avoid this, we divided each DNA strand into length segments randomly drawn from a predefined set of lengths, ℓ_i, such that no piece of strand is used twice and the whole strand is used. The set is given by $\ell_i = \frac{\ell_{max}}{n} i$ with $0 < i \leq n$, ℓ_{max} the maximum segment length and n the number of different segments lengths; here $n = 20$ and $\ell_{max} \approx 120$ nm. This yields a logarithmic length distribution of the segment lengths. In Fig. 9.2(d)) we show the number of segments for the different lengths. The same procedure is applied to compute the kurtosis (Figs. 9.2 and 9.4) and the bending angle histograms (Fig. 9.5). For the latter we used $n = 3$ and $\ell_{max} = L$. Furthermore, we investigated the influence of pre-selecting non self-intersecting molecules only for the analysis. For a persistence length of $P = 18.7$ nm less than 4% of 474 bp long molecules are expected to be self intersecting (for $P = 52.7$ nm less than 1%). That is why we restrict the fitting to $L \in [0, 2P]$. Without this restriction, i.e. fitting $\langle \theta^2 \rangle$ and $< \cos\theta >$ over the entire range, would yield a slight overestimation of P. The error bars given in the Figs. 9.2 and 9.5 show one standard deviation which is computed from 20 different random draws ℓ_i from the same set of molecules. Thus the error bars are an indication for the statistical reliability of that measurement point. For the kurtosis and $\langle D^4 \rangle$ we applied error propagation to determine the standard deviation. For the bending angle distribution $-\ln G(\theta, l)$ we include the cumulative probability by evaluation of the error function for the bins in the deflection angle. The offset of the distribution is not neglected, see Eq. 9.12. Here a remark has to be made regarding the estimation of the bending angle. The DNA molecules are not imaged infinitely sharp but blurred due to the inherent width of DNA of ~ 2 nm and the point spread function of the imaging system. This effective blurring introduces correlation in the estimated tangents and thus angles over short separation ($L \approx \sigma$). From the evaluation on test images we found that this correlation introduces an apparent shorting of the strand when evaluating $\theta(L)$. The amount of apparent shorting was found to be equal to the standard deviation of the Gaussian spot placed at the simulated coordinate. As a consequence, a correction transformation of $L \to L + \sigma$ in the evaluation of $\theta(L)$ was applied. Obviously, this effect is strongest for small $L \approx \sigma$ and negligible for $L \gg \sigma$. From EM data we estimated the effective blurring to correspond to a Gaussian of $\sigma = 1$ nm. The evaluation of $R(L)$ does not suffer from this as the center point is not influenced by blurring.



(a) (d)

(b) (e)

(c)

Figure 9.10: Example of the data analysis of a Monte-Carlo simulation of 500 heteropolymer DNA molecules (474 bp long) with $P = 150$ bp (51.0 nm), $\theta_{stat} = 5.3°$ and $p = 0.5$. Data are shown for i) all simulated strands (×) and ii) only the non self-intersecting strands (○). a)-b) show estimates of the persistence length based on $\langle \cos\theta(L) \rangle$ Eq. 9.2 and $\left\langle R^2(L) \right\rangle$ Eq. 9.1 respectively; c) shows the negative logarithm $-\ln G$ Eq. 9.17 of the occurrence of the deflection angle θ for $L = 4$ and $L = 6$ nm; d) shows the normalized difference $(\left\langle R^4 \right\rangle - \left\langle R^2 \right\rangle^2)/L^4$ Eq. 9.9 with fits for the 2D Eq. 9.8 and 3D Eq. 9.7 case; e) shows the kurtosis $\left\langle \theta^4 \right\rangle / \left\langle \theta^2 \right\rangle^2$ (which is equal to 3 for θ Gaussian distributed around zero).

The experimentally found kurtosis as a function of contour length (see Fig. 9.2) was compared to Monte Carlo simulations with static bending θ_0 on a comparable number of molecules as presented in the experiments (see Fig. 9.4). Also for this simulations a non-overlapping logarithmic sampling of the polymers segments is essential to judge the statistical spread in the result. Non-uniform probability, i.e. $p \neq 0.5$, for left and right static bending was inspected in the same way, as well as an alternative hypothesis $p = 0, \theta = 0$.

9.9 Derivation of the forth order moments of the end-to-end distance in 2D

The energy required to bend a polymer molecule in two dimensions, i.e. a flexible rod, is given by [77, §17-18]:

$$E = \frac{\Upsilon\, I\, \theta^2}{2L} \tag{9.15}$$

where θ is the angle between the tangent vectors to the rod separated by a distance l along the polymer. The macroscopic quantities Young's modulus Υ and area moment of inertia I of the molecule are related to the persistence length $P \equiv P_{3D}$ [77] by $\Upsilon\, I = k_B T P$ where k_B is the Boltzmann constant and T is the absolute temperature. The Boltzmann distribution of states is given by:

$$\mathcal{P}(E) = \frac{g(E)\, e^{-\frac{E}{k_B T}}}{\int_0^\infty g(E')\, e^{-\frac{E'}{k_B T}} dE'}\,. \tag{9.16}$$

In our case the density of states $g(E)$ is constant and as such can be dropped from the equation. Combining the above equations gives the normalized probability distribution function for the bend angle θ:

$$G(\theta, L) = \frac{e^{-\frac{\theta^2 P}{2L}}}{\int_{-\infty}^{\infty} e^{-\frac{s^2 P}{2L}} ds} = \sqrt{\frac{P}{2\pi\, L}} e^{-\frac{P\,\theta^2}{2\,L}}\,. \tag{9.17}$$

We must apply binning to calculate the histogram $-\ln G(\theta, L)$ from the measurements. For a bin size of $\Delta\theta$ around a bin center θ we obtain

$$G(\theta, L) = \operatorname{erf}\left(\sqrt{\frac{P}{2L}}\left(\theta + \frac{\Delta\theta}{2}\right)\right) - \operatorname{erf}\left(\sqrt{\frac{P}{2L}}\left(\theta - \frac{\Delta\theta}{2}\right)\right)\,. \tag{9.18}$$

As we investigate the magnitude of θ a factor of 2 enters the above equation. Neglecting the binning effect simplifies the formula to $-\ln G(\theta, L) \propto \frac{P}{L}\theta^2$ which is most often seen in literature. However, such a plot cannot be used to judge the quality of the persistence length fit. The average moments of the bend angle are given by

$$\begin{aligned} \langle \theta^n \rangle &= \int_{-\infty}^{\infty} \theta^n\, \mathcal{P}(\theta)\, d\theta \\ &= 2^{(n-2)/2}\,(1 + (-1)^n)\left(\frac{L}{P}\right)^{n/2} \Gamma\left(\frac{1+n}{2}\right) \end{aligned} \tag{9.19}$$

As our probability density function is even all odd moments for θ are zero, i.e. $\langle \theta^n \rangle = 0$ for $n = 2k{+}1$ and $k \in \mathbb{N}^+$. Note that the Gamma function for half integer arguments is given by:

$$\Gamma(n + \frac{1}{2}) = \frac{(2n)!\sqrt{\pi}}{n!\, 2^{2n}} \tag{9.20}$$

Combining Eq. 9.19 and Eq. 9.20 gives the even moments, i.e. $n = 2k$ and $k \in \mathbb{N}^+$;

$$\langle \theta^n \rangle = \frac{n!}{(n/2)!}\left(\frac{L}{2\,P}\right)^{n/2} \tag{9.21}$$

as such $\langle \theta^2 \rangle = \frac{L}{P}$ and $\langle \theta^4 \rangle = \frac{3L^2}{P^2}$.
The first two moments of the cosine of the bend angle are given by:

$$\begin{aligned} \langle \cos\theta \rangle &= \int_{-\infty}^{\infty} \cos\theta\, \mathcal{P}(\theta)\, d\theta \\ &= e^{-\frac{L}{2\,P}} \end{aligned} \tag{9.22}$$

$$\begin{aligned} \langle \cos^2\theta \rangle &= \int_{-\infty}^{\infty} \cos^2\theta\, \mathcal{P}(\theta)\, d\theta \\ &= \frac{1}{2}\left(1 + e^{\frac{-2\,L}{P}}\right) \end{aligned} \tag{9.23}$$

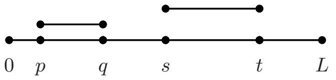
where both L and P are positive.

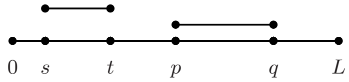
The Euclidean distance between points s and t on the curve separated by a distance L is given by $\boldsymbol{R} = \int_0^L \boldsymbol{u}(s)\, ds$ with $\boldsymbol{u}(s) = [\cos(\theta_s), \sin(\theta_s)]^T$ the tangent vector to the curve at contour position s. Now the average second order moment $\langle R^2 \rangle$ is given by:

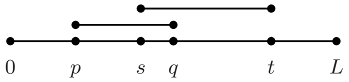
$$\begin{aligned} \langle R^2 \rangle &= \int_0^L \int_0^L \langle \boldsymbol{u}(s) \cdot \boldsymbol{u}(t) \rangle\, dt\, ds \\ &= \int_0^L \int_0^L \langle \cos\theta_s \cos\theta_t + \sin\theta_s \sin\theta_t \rangle\, dt\, ds \\ &= 2\int_0^L \int_0^t \langle \cos\left(\theta_s - \theta_t\right) \rangle\, ds\, dt = 2\int_0^L \int_0^t e^{-\frac{t-s}{2P}}\, ds\, dt \\ &= 4\,PL - 8\,P^2 + 8\,P^2 e^{-\frac{L}{2P}}\ . \end{aligned} \tag{9.24}$$

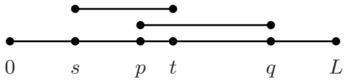
Note that the argument of the average cosine is always positive, as the length is always positive. With the average operator defined as $\langle \cdot \rangle = \int_{-\infty}^{\infty} \cdot\, G(\theta, L) d\theta$ the average fourth order moment $\langle R^4 \rangle$ can be derived in a similar fashion:

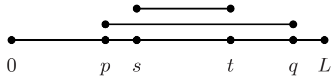
$$
\begin{aligned}
\langle R^4\rangle =&\langle \int_0^L \boldsymbol{u}(s)\,ds\cdot\int_0^L \boldsymbol{u}(t)\,dt\int_0^L \boldsymbol{u}(p)\,dp\cdot\int_0^L \boldsymbol{u}(q)\,dq\rangle\\
=&\int_0^L\int_0^L\int_0^L\int_0^L \langle \boldsymbol{u}(s)\cdot\boldsymbol{u}(t)\,\boldsymbol{u}(p)\cdot\boldsymbol{u}(q)\rangle\,ds\,dt\,dp\,dq\\
=&\int_0^L\int_0^L\int_0^L\int_0^L \langle\cos(\theta_s-\theta_t)\cos(\theta_p-\theta_q)\rangle\,ds\,dt\,dp\,dq\\
=&4\Big(\int_0^L\int_0^t\int_0^s\int_0^q \langle\cdot\rangle_{A1}\,dp\,dq\,ds\,dt+\\
&\int_0^L\int_s^L\int_t^L\int_p^L \langle\cdot\rangle_{A2}\,dq\,dp\,dt\,ds+\\
&\int_0^L\int_0^t\int_s^t\int_0^s \langle\cdot\rangle_{B1}\,dp\,dq\,ds\,dt+\\
&\int_0^L\int_0^t\int_s^t\int_t^L \langle\cdot\rangle_{B2}\,dq\,dp\,ds\,dt+\\
&\int_0^L\int_0^t\int_t^L\int_0^s \langle\cdot\rangle_{C1}\,dp\,dq\,ds\,dt+\\
&\int_0^L\int_0^t\int_s^t\int_s^q \langle\cdot\rangle_{C2}\,dp\,dq\,ds\,dt\Big)\\
=&8\Big(\int_0^L\int_0^t\int_0^s\int_0^q \langle\cdot\rangle_{A1}\,dp\,dq\,ds\,dt+\\
&\int_0^L\int_0^t\int_s^t\int_0^s \langle\cdot\rangle_{B1}\,dp\,dq\,ds\,dt+\\
&\int_0^L\int_0^t\int_t^L\int_0^s \langle\cdot\rangle_{C1}\,dp\,dq\,ds\,dt\Big)
\end{aligned}
\tag{9.25}
$$

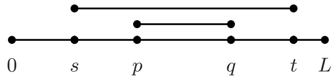












To handle the correlation between the different integration variables, the integration is separated in to six parts (see above). In case A we have non overlapping segments pq and st whereas in respectively case B these segments are partially overlapping.

$$
\begin{aligned}
\langle\cdot\rangle_{A1} &= \langle\cos(\theta_s-\theta_t)\cos(\theta_p-\theta_q)\rangle\\
&= \langle\cos(\theta_s-\theta_t)\rangle\langle\cos(\theta_p-\theta_q)\rangle\\
&= e^{-\frac{t-s}{2P}}e^{-\frac{q-p}{2P}}
\end{aligned}
\tag{9.26}
$$

$$
\begin{aligned}
\langle\cdot\rangle_{B1} &= \langle\cos(\theta_s-\theta_t)\cos(\theta_p-\theta_q)\rangle\\
&= \langle\cos(\theta_s-\theta_p)\rangle\langle\cos(\theta_q-\theta_s)\cos(\theta_q-\theta_s)\rangle\langle\cos(\theta_t-\theta_q)\rangle\\
&= \langle\cos(\theta_s-\theta_p)\rangle\langle\cos^2(\theta_q-\theta_s)\rangle\langle\cos(\theta_t-\theta_q)\rangle\\
&= e^{-\frac{s-p}{2P}}e^{-\frac{t-q}{2P}}\frac{1}{2}\left(1+e^{-\frac{2(q-s)}{P}}\right)
\end{aligned}
\tag{9.27}
$$

$$
\begin{aligned}
\langle\cdot\rangle_{C1} &= \langle \cos\left(\theta_s - \theta_t\right) \cos\left(\theta_p - \theta_q\right)\rangle \\
&= \langle \cos\left(\theta_s - \theta_p\right) \cos\left(\theta_t - \theta_s\right) \cos\left(\theta_t - \theta_s\right)\rangle \langle \cos\left(\theta_q - \theta_t\right)\rangle \\
&= \langle \cos\left(\theta_s - \theta_p\right)\rangle \langle \cos^2\left(\theta_t - \theta_s\right)\rangle \langle \cos\left(\theta_q - \theta_t\right)\rangle \\
&= e^{-\frac{s-p}{P}} e^{-\frac{q-t}{P}} \frac{1}{2}\left(1 + e^{-\frac{2(t-s)}{P}}\right)
\end{aligned}
\tag{9.28}
$$

Combining these equations we obtain

$$
\langle R^4 \rangle = 32\, L^2 P^2 - 240\, L P^3 + 696\, P^4 - \frac{320}{3} L P^3 e^{-\frac{L}{2P}} - \frac{6272}{9} P^4 e^{-\frac{L}{2P}} + \frac{8}{9} P^4 e^{-\frac{2L}{P}} \,. \tag{9.29}
$$

9.10 Derivation of the kurtosis for the heteropolymer model

For the purposes of analysis we make the following assumptions. First, we used the Schellman [122] approach for the description of a DNA chain. Second, Schellman also suggested that a bending potential can still be quadratic for the non-zero value of static bend. This idea was later used by Cognet, leading to the following formula of the bending potential

$$
p(\theta_i) = N^{-1/2} e^{-\frac{g_i'' \theta_i^2}{2RT}} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\left(\theta_i - \theta_{i,stat}\right)^2}{2\sigma^2}} \,. \tag{9.30}
$$

Third, Schellman showed that a good approximation to the shape of real polymers can be achieved by using the following formula for the persistence length $P \sim 1/(1 - \langle \cos\theta_{i,dyn} \rangle \langle \cos\theta_{i,stat} \rangle)$ [124], where $\langle \cos\theta_{i,dyn} \rangle$ accounts for the dynamic behavior of polymer chains and $\langle \cos\theta_{i,stat} \rangle$ for local static bending. The latter means averaging of all static bending angles, i.e. along DNA position. In another words, it requires double averaging. First, over an ensemble of molecules and second along the length of the molecules. In view of small values for both $\langle \theta_{i,dyn} \rangle$ and $\langle \theta_{i,stat} \rangle$ (usually less than about 5 degrees) we use the following formula [20]

$$
P \approx 2/\left\langle \theta^2 \right\rangle \,, \tag{9.31}
$$

where $\langle \theta^2 \rangle = \left\langle \theta_{dyn}^2 \right\rangle + \langle \theta_{stat}^2 \rangle$ [124]. Next, we have to keep in mind that according to the measurement procedure all values for $\langle \cos\theta \rangle$ etc. are obtained using double averaging, i.e. configurational and positional. To start the analysis let us consider a segment of polymer chain with fixed first and last points (fixed length along the DNA). This is configurational averaging similar to [48] with the difference that static bending angles in plane are included. In the following we use short $\theta^0 \equiv \theta_{stat}$ and $\sigma \equiv \theta_{dyn}$. The angel θ_n over a length along the DNA n is

$$
\theta_n = \sum_1^n \theta_i = \sum_1^n (\theta_i - \theta_i^0 + \theta_i^0) = \sum_1^n (\theta_i - \theta_i^0) + \sum_1^n \theta_i^0 = \sum_1^n \Delta_i + \sum_1^n \theta_i^0 \,, \tag{9.32}
$$

with $\Delta_i = \theta_i - \theta_i^0$ and since $\left(\sum_1^n \Delta_i\right)^2 = \sum_1^n \Delta_i^2 + \sum_{i\neq j} \Delta_i \Delta_j$. Using the harmonic potential of $(\theta_i - \theta_i^0)$ as in eq.(9.30) we have $\langle \Delta_i^2 \rangle = \sigma^2$ and $\langle \Delta_i^4 \rangle = 3\sigma^4$. From that follows

$$\left\langle \left(\sum_1^n \Delta_i\right)^2 \right\rangle = \sum_1^n \left\langle \Delta_i^2 \right\rangle + \sum_{i\neq j} \left\langle \Delta_i \right\rangle \left\langle \Delta_j \right\rangle = n\sigma^2 , \tag{9.33}$$

$$\left\langle \left(\sum_1^n \Delta_i\right)^4 \right\rangle = \sum_1^n \left\langle \Delta_i^4 \right\rangle + 3\sum_{i\neq j} \left\langle \Delta_i^2 \right\rangle \left\langle \Delta_j^2 \right\rangle + \underbrace{4\sum_{i\neq j} \left\langle \Delta_i \right\rangle \left\langle \Delta_j^3 \right\rangle}_{=0} \tag{9.34}$$

$$= 3n\sigma^4 + 6\frac{n(n-1)}{2}\sigma^4 = 3n^2\sigma^4 . \tag{9.35}$$

Let us introduce $A_n^0 = \sum_1^n \theta_i^0$ for the intrinsic bending over length n. Note that this is identical for all molecules for a fixed length along the DNA. Thus we can write for the moments

$$(\theta_n)^2 = \left(\sum_1^n \Delta_i\right)^2 + \left(A_n^0\right)^2 + 2A_n^0 \sum_1^n \Delta_i \tag{9.36}$$

$$(\theta_n)^4 = \left(\sum_1^n \Delta_i + A_n^0\right)^4 \tag{9.37}$$

and finally

$$\left\langle (\theta_n)^2 \right\rangle = n\sigma^4 + \left(A_n^0\right)^2 \tag{9.38}$$

$$\left\langle (\theta_n)^4 \right\rangle = 3n^2\sigma^4 + 6\left(A_n^0\right)^2 n\sigma^2 + \left(A_n^0\right)^4 . \tag{9.39}$$

It means for a fixed length along the DNA we get for the kurtosis

$$k = 3 - \frac{2(A_n^0)^4}{(n\sigma^2 + (A_0^n)^2)^2} . \tag{9.40}$$

It is clear that the kurtosis is always smaller than 3 for any sequence exhibiting static bending with $A_n^0 \neq 0$.

For our purposes we have to make the second averaging over the position, which means that we have to average the higher moments of static angles, this leads to the following formulas

$$k = \frac{\langle\langle \theta_n^4 \rangle\rangle}{\langle\langle \theta_n^2 \rangle\rangle^2} \tag{9.41}$$

$$\left\langle \langle \theta_n^2 \rangle \right\rangle = n\sigma^4 + \left\langle (A_n^0)^2 \right\rangle \tag{9.42}$$

$$\left\langle \langle \theta_n^4 \rangle \right\rangle = 3n^2\sigma^4 + 6\left\langle (A_n^0)^2 \right\rangle n\sigma^2 + \left\langle (A_n^0)^4 \right\rangle . \tag{9.43}$$

To simplify the calculations let us make the assumption that all static bends occur in plane and behave independent of each other. Moreover, we consider that all static bends are equal in value, i.e. $|\theta_i^0| = |\theta^0|$. Let us also assume that the choice of sign is random (plus or minus direction have equal probability of $p = 0.5$), implying that the behavior of a "frozen" polymer chain (no dynamic fluctuations) resembles a one dimensional walk in terms of angles. From this it follows that

$$\begin{aligned} \left\langle (A_n^0)^2 \right\rangle &= n(\theta^0)^2 && (9.44) \\ \left\langle (A_n^0)^4 \right\rangle &= (\theta^0)^4(3n^2 - 2n)\,. && (9.45) \end{aligned}$$

Inserting eqs.(9.44,9.45) into eqs.(9.42,9.43) yields

$$\begin{aligned} \left\langle \langle \theta_n^2 \rangle \right\rangle &= n\sigma^2 + n(\theta^0)^2 && (9.46) \\ \left\langle \langle \theta_n^4 \rangle \right\rangle &= 3n^2\sigma^4 + 6n(\theta^0)^2 n\sigma^2 + (\theta^0)^4(3n^2 - 2n) && (9.47) \end{aligned}$$

and finally for the kurtosis k

$$k = 3 - \frac{2}{n}\frac{(\theta_0)^4}{(\sigma^2 + (\theta^0)^2)^2}\,. \qquad (9.48)$$

Conclusions

10

This thesis deals with two image analysis topics. The first topic is orientation estimation in multimodal regions, i.e. measuring the orientation of the constituents of a complex region. The second topic adresses the problem of centreline extraction of curvilinear structures for accurate shape analysis. Both tasks are closely related to the task of orientation estimation. Although obvious for the first task, for the latter task one has to realise that one can derive the shape properties from the centreline of curvilinear objects and vice versa.

10.1 Multi Orientation estimation

This thesis shows several methods to determine the orientation of the constituents in multi modal regions. At the core of most of these methods lies the task of disentangling the orientation information and the subsequent assignment to the respective modes. We approach the problem in several distinct ways, each with its own benefits and shortcomings on criteria such as computational complexity, memory requirements, determined outcome and interpretability. This ranges from the sometimes hard to interpret outcome of the memory expensive filterbank methods to the less determined outcome of the clustering based methods.

We started out with the extension of the orientation space concept to 3D, Chapter 2. To this end a 3D filterbank was constructed with dedicated orientation selective filters for lines and planes. This resulted in a $3D \rightarrow 5D$ mapping which basically adds a 2D orientation histogram, containing the evidence for a particular orientation, to each location in the 3D image. This method can solve some inherently difficult tasks like separating crossing objects as well as orientation measurements in complex regions. We had to create a new set of tunable $3D$ orientation selective filters as well as a multi resolution grid for sampling. Furthermore, we introduced a novel way to visualise the resulting $5D$ space. In a brute-force implementation, the memory requirements can become huge i.e. approximately the number of filters used times the size of the original image. Although increasing the computational complexity, the method allows for a local approach where the orientation histogram is only constructed in points of interest. The locality can be employed in a spatio-orientation multi-resolution approach. Increasing the filter footprint increases the orientation resolution (orientation selectivity) while sacrificing spatial resolution (the localisation).

Chapter 3 describes a method for the detection and orientation characterisation the constituents of crossing lines and saddle points. The method is based on the following observation: the structure tensor gives rise to saddle points in the orientation map when applied to line crossings and (skewed) saddle points. This observation is easily understood assuming the orientation of the average edge in the analysis window to be assigned to each pixel, the averaging nature of the structure tensor will cause smooth transitions between uniform orientation regions. A second derivative method is used for the detection, localisation and orientation estimation of these saddle points in the orientation map. Based on this information the local neighbourhood of each saddle point is segmented in two regions based on the bowtie in the saddle points. The segmented regions are analysed by means of the structure tensor. Although only applicable to a small subset of structures it can be a fast and powerfull tool in determining e.g. projections and distortions by means of recorded checkerboard patterns.

In Chapter 4 we present a novel method for multi-orientation detection and analysis based on streamlines. Here the orientation estimates in multimodal regions are obtained by measuring the orientation of the constituents in regions where there is almost no overlap or mixing. The connection between the unimodal and mixed regions is established by means of streamlines which follow the local orientation of the structure. Although only applicable to specific problems this method shows a good angular selectivity and low memory requirements. Furthermore, the method relies on the assumption of a high correspondence for the constituents in the mixed and the respective unimodal regions. This assumption is true for straight lines but is obviously flawed for curvilinear structures. This said we have given an estimate of the region of interest needed to disentangle the different contributions which indicated that mildly curved structures should not deteriorate the results to much.

The methods presented in Chapter 5 and Chapter 6 both combine pattern recognition with image analysis techniques. These methods decompose the structure tensor to allow analysis and subsequent representation of multiple oriented structures inside a local neighbourhood. Both methods rely on the fact that oriented structures will give rise to clustered regions in gradient space, i.e. points with similar orientations will be mapped to a single axis in gradient space. To solve the ambiguity in the gradient mapping, a transformation is used which maps antipodal vectors to the same point. This results in compact clusters for points with the same orientation. These clusters are then analysed by means of the k-means clustering algorithm. The resulting clusters are analysed separately by the structure tensor. The method presented in Chapter 5 gives promissing results although spatial connectivity between the points in the clusters is not required. The second method, i.e. Chapter 6, takes the spatial connectivity into account by introducing a distance measure which allows for the assignment of each pixel to the purest point in the local neighbourhood. Pure in the sense that is originates from a single orientation mode. This point is assumed to be representative for all points associated with it. This segmentation based on the pureness of each point ensures that only spatially extended regions are taken into account and spatially uncorrelated points can be suppressed in the analysis. Finally, clustering of these spatially connected regions results in a mostly accurate segmentation of the complex neighbourhood, after which the segmented regions can be analysed separately. These methods are reasonably fast, robust, and offer an excellent alternative to the computationally and memory expensive filterbank methods. In this thesis we used a hard labelling of points which in some cases gives rise to a small bias. To correct for this, a soft labelling could be applied.

10.2 Distance transforms and path extraction for shape analysis of curvilinear structures

The second part of this thesis starts out with a review of a selection of distance transforms, Chapter 7. Although not initially intended to be applied to small curvilinear structures we show that all existing methods cut the corner and do not accurately delineate the centreline. We noted that the Euclidean, grey-weighted and fast-marching distance transforms all rely on the Huygens principle and as such the wave propagation should locally be modelled as a spherical wave. As such we introduced the circular wave as a propagation model. Although no closed form solution was found, the new method results in a more accurate distance map, i.e. compared to the aforementioned methods. Furthermore, it gives exacts results on a flat cost function.

Although optimising distance transforms could possibly yield the true minimum cost path, it does not give the true centreline of curvilinear structures. A minimum costpath will always cut the corners and hence produce a smoothed version of the true centreline. An iterative solution is presented in Chapter 8 to reduce this effect for these structures. The proposed method is based on the observation that distance transforms can yield the true centreline for linear objects when their curvature is negligible. As such, by iteratively warping the cost function such that the minimum cost path becomes a straight line, we are able to eliminate the cutting of corner effect. The obtained paths after applying the inverse transformation give highly accurate and precise estimates of the centreline. In some cases a damping term is necessary to prevent small oscillations.

In Chapter 9 we apply the developed iterative minimum cost path algorithm to a large set of DNA molecules prepared under different conditions and imaged with transmission electron microscopy and atomic force microscopy. The analysis of long polymers like DNA with these microscopy techniques requires the deposition of the molecules on a surface in a dried state. This however does not ensure the dimensionality of the conformational state of the molecules. The molecules could be adhered in an equilibrium state, i.e. a 2D state, or the state can be frozen at deposition leading to a 3D conformational state or even an intermediate state. Our data shows that the molecules are adhered in a 2D dimensional state for different preparation conditions and distances up to approximately 120 nm, although with different persistence lengths for each condition. We argue that the difference in persistense length is due to distortions in the DNA backbone which can be described by an additional static bending at each nucleitide step of approximately 2°. Where the additional DNA bending (kinking) is assumed to be introduced by the deposition of the DNA onto a charged imaging surface, i.e. carbon film or polylysine.

Isophote curvature

A

The normalised gradient of an image I is given by

$$\boldsymbol{g} = \frac{\nabla I}{|\nabla I|} \tag{A.1}$$

The basis defined by the isophote curvature, $\boldsymbol{B}_{iso}$, is given by the normalised gradient vector $\boldsymbol{g}$ and the two principal curvature vectors, $\boldsymbol{v}_{\kappa_1}$ and $\boldsymbol{v}_{\kappa_2}$, i.e.

$$\boldsymbol{B}_{iso} = (\boldsymbol{g}, \boldsymbol{v}_{\kappa_1}, \boldsymbol{v}_{\kappa_2}) \tag{A.2}$$

The Hessian of I is given as

$$\boldsymbol{H} = \begin{pmatrix} \frac{\partial^2 I}{\partial x \partial x} & \frac{\partial^2 I}{\partial x \partial y} & \frac{\partial^2 I}{\partial x \partial z} \\ \frac{\partial^2 I}{\partial y \partial x} & \frac{\partial^2 I}{\partial y \partial y} & \frac{\partial^2 I}{\partial y \partial z} \\ \frac{\partial^2 I}{\partial z \partial x} & \frac{\partial^2 I}{\partial z \partial y} & \frac{\partial^2 I}{\partial z \partial z} \end{pmatrix} \tag{A.3}$$

Now we can choose an orthonormal basis with respect to the normalised gradient

$$\boldsymbol{B}_g = (\boldsymbol{g}, \boldsymbol{u}, \boldsymbol{v}) \tag{A.4}$$

where the vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ should be orthogonal and span the tangent plane to the local structure in the image defined by $\boldsymbol{g}$. This basis can be used to rotate the Hessian $\boldsymbol{H}$ in such a way that the first basis vector coincides with the gradient direction.

$$\boldsymbol{H}' = \boldsymbol{B}_g^T \boldsymbol{H} \, \boldsymbol{B}_g = \left(\begin{array}{c|cc} \frac{\partial^2 I}{\partial g \partial g} & \cdot & \cdot \\ \hline \cdot & \frac{\partial^2 I}{\partial u \partial u} & \frac{\partial^2 I}{\partial u \partial v} \\ \cdot & \frac{\partial^2 I}{\partial v \partial u} & \frac{\partial^2 I}{\partial v \partial v} \end{array} \right) \tag{A.5}$$

the elements on the dots are not shown as they are not of interest for the calculation of the isophote curvatures. Note that there is one degree of freedom left as $\boldsymbol{u}$ and $\boldsymbol{v}$ can be rotated freely around the gradient axis. Now $\boldsymbol{u}$ and $\boldsymbol{v}$ are rotated in such a way that the resulting vectors coincide with the principal curvature vectors. This is done by extraction the indicated 2×2 Hessian sub matrix from $\boldsymbol{H}'$. And performing an eigensystem analysis on this Hessian matrix. The eigenvalues, λ_1 and λ_2, of this analysis are connected to the principal curvatures, κ_1 and κ_2, by the gradient magnitude:

$$\kappa_i = -\frac{\lambda_i}{|\nabla I|}\,. \tag{A.6}$$

And the resulting 2D eigenvectors, $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, correspond with the principal curvature directions. Padding these eigenvectors with a zero in front gives the principal curvature vectors in the 3D basis $\boldsymbol{B}_g$, i.e.

$$\boldsymbol{B}_{g'} = (\boldsymbol{g}, (0, \boldsymbol{v}_1^T)^T, (0, \boldsymbol{v}_2^T)^T) \tag{A.7}$$

Returning to the original basis the isophote basis is given by

$$\boldsymbol{B}_{iso} = \boldsymbol{B}_g \, \boldsymbol{B}_{g'} \tag{A.8}$$

Summary

The orientation of curvilinear structures contains a wealth of information, which is often only accessible after employing advanced image processing techniques. This thesis presents solutions for two important problems when dealing with orientation analysis in digital images: multi-valued orientation analysis for neighbourhoods composed of multiple structures with different orientations and methods for the automated extraction and analysis of curvilinear structures under noisy conditions. The latter is applied to measure the flexibility of DNA under a range of experimental conditions.

The first part of this thesis deals with orientation analysis in complex regions. Complex or multi-modal neighbourhoods are regions that are composed of multiple unimodal (simple) structures, i.e. lines and edges each with their own amplitude, scale and orientation. Traditionally, orientation can only be measured accurately in simple neighbourhoods. Most existing algorithms will consider a complete neighbourhood and measure some sort of average orientation. In this thesis we have developed methods which can be applied to complex neighbourhoods. As such the main challenge for orientation analysis in multi-modal regions is to disentangle the orientation information, followed by orientation measurements in the resulting unimodal datasets. As multiple structures interfere during image formation, the orientation information mixes as well. Note that many advance image filters for noise suppression exaggerate this effect and make the task even more difficult.

We tackled this problem by inventing novel filtering and analysis schemes ranging from generic to more application specific. We present a filterbank with rotated versions of a 3D orientation selective filter for lines, planes and edges in 3D. Here the filterbank scans the orientation dimensions and the filters capture the orientation energy of structures within the selected orientation band. To avoid a data explosion as in the filterbank approach, we use streamlines to convey orientation information from unimodal regions into mixed regions like junctions and crossings. Alternatively we present two methods to cluster the gradient structure tensor in a complex neighbourhood, either with or without enforcing spatial relations in the local neighbourhood.

In the second part of this thesis we investigated the possibility of automated shape extraction and analysis of curvilinear structures, with special interest in the extraction and statistical shape properties of DNA molecules imaged by transmission electron microscopy (TEM) and atomic force microscopy (AFM). First we present the tools to extract the centreline of curvilinear structures based on minimum cost paths. These methods are based on distance transforms employing an expanding wave front but use different models to describe the expansion. We present a new distance transform which models the wavefront locally by a circular rather than a planar wave which increases the accuracy and precision with respect to existing methods.
Although optimising of distance transforms could yield a true minimum cost path, it does not correspond to the centreline of curvilinear structures. Minimum cost path algorithms will always minimise the total cost by cutting corners. We have therefore introduced a novel iterative procedure which does not suffer from this effect and yields highly accurate and precise estimates of the centreline of curvilinear structures, even under noisy conditions. This method is based on the observation that the cutting the corner effect is non existent for linear objects when their curvature is negligible.
Finally we apply this new procedure to analyse the flexibility of DNA near charged surfaces. The analysis of DNA with AFM and TEM requires that the molecules are deposited on a surface in a dried state. This does however not ensure the dimensionality of the molecules. Our data shows that the molecules are adhered in a 2D state for different preparation conditions and distances up to approximately 120 nm, although with different persistence lengths for each condition. We argue that the difference in persistence length is due to distortions in the DNA backbone which can be described by an additional static bending (kinking) at each nucleotide step of approximately 2 degrees. This additional DNA bending is assumed to be introduced by the deposition of the DNA onto a charged imaging surface, i.e. carbon film or polylysine.

Samenvatting

De oriëntatie van gekromde lineaire structuren bevat een weelde aan informatie die vaak alleen toegankelijk wordt na toepassing van geavanceerde beeldbewerkingstechnieken. Dit proefschrift behandelt oplossingen voor twee belangrijke problemen in het kader van oriëntatie analyse in digitale beelden: meervoudige oriëntatie analyse in omgevingen samengesteld uit meerdere structuren met verschillende oriëntaties en de automatische extractie en analyse van gekromde lineaire structuren onder sterke ruiscondities. Waar de laatste is toegepast om de flexibiliteit van DNA te bepalen onder een reeks van experimentele condities.

Het eerste deel van dit proefschrift behandelt oriëntatie analyses in complexe omgevingen. Complexe of multimodale omgevingen zijn regio's die zijn samengesteld uit meerdere unimodale structuren, bijvoorbeeld lijnen en randen elk met hun eigen amplitude, schaal en oriëntatie. Oriëntatie kan normaalgesproken alleen nauwkeurig gemeten worden in enkelvoudige omgevingen. De meeste bestaande methoden beschouwen een omgeving als geheel en zullen als gevolg een min of meer gemiddelde oriëntatie meten. In dit proefschrift hebben we methoden ontwikkeld die kunnen worden toegepast op complexe omgevingen. De belangrijkste stap bij oriëntatiemetingen in complexe omgevingen is het ontwarren van de oriëntatie informatie, gevolgd door metingen in de gevonden unimodale data. Deze taak wordt bemoeilijkt doordat complexe structuren interfereren gedurende de beeldacquisitie waardoor de oriëntatie informatie gemengd raakt. De meeste geavanceeerde ruisonderdrukkende filters zullen dit effect versterken.

Dit probleem hebben we aangepakt door het ontwikkelen van nieuwe filter- en analyseschema's, variërend van generiek tot applicatie specifiek. We presenteren een filterbank met geroteerde versies van een 3D oriëntatie selectief filter voor lijnen, vlakken en randen in 3D. Deze filterbank scant de oriëntatie dimensies waarbij de filters de oriëntatie energie meten van de structuren binnen de geselecteerde oriëntatie band. Om een data explosie zoals in de filterbankmethode te voorkomen, introduceren we een op stroomlijnen gebaseerde methode om oriëntatie informatie te transporteren van unimodale regio's naar gemengde regio's zoals splitsingen en kruisingen. Ook presenteren we twee methoden om de gradiënt structuurtensor te clusteren in een complexe omgeving. Enerzijds door het niet en anderzijds door het wel meenemen van spatieële relaties in de omgeving.

In het tweede deel van dit proefschrift bestuderen we de mogelijkheid van automatische vormextractie en analyse van gekromde structuren, met als doel de extractie en bepaling van statistische eigenschappen van DNA moleculen afgebeeld met transmissie elektronen microscopie (TEM) en atomic force microscopie (AFM). Eerst introduceren we, gebaseerd op minimum kosten paden methoden, het gereedschap om de middellijn van gekromde structuren te kunnen bepalen. Al deze methoden zijn gebaseerd op afstandstransformaties die gebruik maken van een uitdijend golffront, waarbij gebruik wordt gemaakt van verschillende modellen om de expansie te beschrijven. We stellen een nieuwe afstandstransformatie voor waarin de expanderende golf wordt beschreven door een circulaire in plaats van een vlakke golf. Dit vergroot de nauwkeurigheid en precisie vergeleken met andere methoden. Hoewel het optimaliseren van afstandstransformaties kan leiden tot een exact minimum kostenpad, zal het niet corresponderen met de middellijn van gekromde lineaire structuren. Minimum kosten paden methoden zullen altijd de kosten proberen te minimaliseren door het afsnijden van bochten. We hebben daarom een nieuwe iteratieve procedure ontwikkeld die geen last heeft van dit effect en zeer precieze schattingen geeft voor de middellijn van gekromde structuren, zelfs onder sterke ruiscondities. Deze methode is gebaseerd op het feit dat het effect van het afsnijden van bochten verwaarloosbaar is voor niet gekromde lineaire structuren.

Tot slot passen we deze nieuwe methode toe om de flexibiliteit van DNA te analyseren in de nabijheid van opgeladen oppervlakken. De analyse van DNA met AFM en TEM vereist dat moleculen op een oppervlak worden opgebracht in een uitgedroogde toestand. Dit legt echter niet de dimensionaliteit van de moleculen vast. Onze gegevens laten zien dat de moleculen worden geadheerd in een 2D toestand voor verschillende preparatietechnieken en lengtes tot ongeveer 120 nm, echter met verschillende persistentie lengtes voor iedere conditie. We stellen dat de verschillen in de gemeten persistentie lengte kan worden toegeschreven aan vervormingen in de hoofdketen van het DNA. Hierbij beschrijven we deze vervorming als een additionele statische buiging (knik) tussen iedere nucleotide van ongeveer 2 graden. We veronderstellen dat deze additionele statische buiging wordt veroorzaakt doordat het DNA wordt opgebracht op een opgeladen oppervlak zoals koolstoffilm of polylisine.

Curriculum vitae

Frank Faas was born in Breda, the Netherlands, on March 15th, 1977. In 1995 he graduated from the O.L.V. Lyceum, in Breda. In the same year he started his astronomy studies at Leiden University. The subject of his master thesis was "The gas temperature in circumstellar disks: effects of dust settling". In 2002 he started his PhD research in the Pattern Recognition Group later renamed to Quantitative Imaging group of the Department of Applied Physics, at Delft University of Technology. This research was funded by FOM in a program called "Multi-Dimensional Measurement Techniques" and carried out under supervision of Prof. dr. ir. Lucas J. van Vliet. During this period he participated in various scientific and educational projects. From 2003-2005 he worked on the Computer Services Labs (CSP), his main task was the teaching of a hands on image processing course. In 2006 he worked for Quintech to automate the recognition of odd size lugage on airport conveyer belts. In 2007 he joined the Electron Microscopy section of the department of Molecular Cell biology at the Leiden University Medical Center. There he is currently working on various image processing and electron microscopy related topics.

Acknowledgements

First and foremost I would like to thank my PhD advisor and mentor Lucas van Vliet, always interested in new ideas, available for discussion and showing trust in a good end. Thanks for the freedom to pursue my own ideas and interests.

Thanks, Cris, Mike and Bernd for making me immediately feel at home at PH and getting me started, Kees for all discussions and runs, Wouter and Jurjen for many a pool game. I enjoyed all people dropping in at our office for a cup of coffee or a talk. To my roommates: Cris, Wouter, Pavel, Vincent and Joost thanks for creating a pleasant (working) atmosphere and keeping up with me.

Joop, thanks for the pleasant and fruitfull cooperation. The vocals of Ronald and the help from Wim and Mandy were also greatly appreciated. Thanks, Piet for all your complot theories, Albert for your humour and Edwin for getting out the frustrations on the water. Thank you all for a pleasant time in Delft!

Bernd, without you this thesis would have looked much different*. More importantly, many thanks for all your help and beeing a great friend.

Pa en Ma dank voor jullie steun en het leren maken van mijn eigen keuzes. Het is fijn te weten dat jullie altijd klaar staan met een luisterend oor en aan me denken, jullie zijn lieverds.

Suus

---

*soms kon ik je achter het behang plakken

Bibliography

[1] T. Aach, C. Mota, I. Stuke, M. Mühlich, and E. Barth. Analysis of superimposed oriented patterns. *IEEE Transactions on Image Processing*, 15(12):3690–3700, 2006. Referred to on p.: 10

[2] J.A. Abels, F. Moreno-Herreo, T. van der Heijden, C. Dekker, and N.H. Dekker. Single-molecule measurements of the persistence length of double-stranded RNA. *Biophysical Journal*, 88:2737–2744, 2005. Referred to on p.: 112

[3] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995. Referred to on p.: 103, 130

[4] M.T. Andersson. *Controllable Multidimensional Filters and Models in Low Level Computer Vision*. PhD thesis, Linköping University, Linköping, Sweden, 1992. Referred to on p.: 11

[5] C.G. Baumann, S.B. Smith, V.A. Blommfield, and C. Bustamante. Ionic effects on the elasicity of single dna molecules. *Proceedings of the National Academy of Sciences of the United States of America*, 94(12):6185–6190, 1997. Referred to on p.: 122, 123

[6] W. Beil. Steerable filters and invariance theory. *Pattern Recognition Letters*, 15: 453–460, 1994. Referred to on p.: 11

[7] J. Bigün. Speed, frequency, and orientation tuned 3-d gabor filterbanks and their design. In *Proceedings of International Conference on Pattern Recognition, ICPR, Jerusalem*, pages C–184–187. IEEE Computer Society, 1994. Referred to on p.: 11, 22

[8] J. Bigün and G.H. Granlund. Optimal orientation detection of linear symmetry. In *Proceedings of the first IEEE International Conference on Computer Vision*, pages 433–438, London, June 8-11 1987. IEEE Computer Society Press. Referred to on p.: 10, 19, 27, 28, 45, 55

[9] J. Bigün, G.H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991. Referred to on p.: 9, 19

[10] H. Blum. A transformation for extracting new descriptors of shape. In Weiant Wathen-Dunn, editor, *Symp. Models Percep. Speech Visual Form*, pages 362–380, Cambridge, 1967. MIT Press. Referred to on p.: 12

[11] G. Borgefors. Distance transformations in arbitrary dimensions. *Computer Vision, Graphics and Image Processing*, 27:321–345, 1984. Referred to on p.: 74, 76

[12] G. Borgefors, I. Nyström, and G. Baja. Computing skeletons in three dimensions. *Pattern Recognition*, 32(7):1225–1236, 1999. Referred to on p.: 13

[13] M. Born and E. Wolf. *Principles of Optics*. Pergamon Press, 6th edition, 1980, 1977. Referred to on p.: 103

[14] W.H. Bosking, Y. Zhang, B. Schonfiled, and D. Fitzpatrik. Orientation selectivity ans the arrangement of horizontal connections in tree shrew striate cortex. *Journal of Neuroscience*, 6:2112–2127, 1997. Referred to on p.: 8

[15] S. Brinkers, H.R.C. Dietrich, F.H. de Groote, I.T. Young, and B. Rieger. The persistence length of double stranded DNA determined using dark field tethered particle motion. *Journal of Chemical Physics*, 130(21):215105, 2009. Referred to on p.: 15

[16] T. Brox, J. Weickert, B. Burgeth, and P. Mrazek. Nonlinear structure tensor. *Image and Vision Computing*, 24:41–55, 2006. Referred to on p.: 10, 27, 70

[17] M. Bussiek, N. Mucke, and J. Langowski. Polylysine-coated mica can be used to observe systematic changes in the supercoiled dna conformation by scanning force microscopy in solution. *Nucleic Acid Research*, 31(22):e137, 2003. Referred to on p.: 112

[18] J. Chen, Y. Sato, and S. Tamura. Orientation space filtering for multiple orientation line segmentation. *Pattern Analysis and Machine Intelligence*, 22(5):417–429, May 2000. Referred to on p.: 11

[19] D.I. Cherny and T.M. Jovin. Electron and scanning force microscopy studies of alterations in supercoiled DNA tertiary structure. *Journal of Molecular Biology*, 313(2):295–307, 2001. Referred to on p.: 123, 124, 127, 128

[20] J.A.H. Cognet, C. Pakleza, D.I. Cherny, E. Delain, and E. le Cam. Static curvature and flexibility measurements of DNA with microscopy. A simple renormalization method, its assessment by experiment and simulation. *Journal of Molecular Biology*, 285(3):997–1009, 1999. Referred to on p.: 112, 119, 121, 138

[21] L.D. Cohen and R. Kimmel. Global minimum for active contour models: A minimal path approach. *International Journal of Computer Vision*, 24(1):57–58, 1997. Referred to on p.: 90

[22] O. Cuisenaire. *Distance transformations: fast algorithm and applications to medical image processing.* PhD thesis, Universite Catholique de Louvain, 1999. Referred to on p.: 13

[23] O. Cuisenaire and B. Macq. Fast euclidean distance transformation by propagation using multiple neighborhoods. *Computer Vision and Image Understanding*, 76(2):163–172, 1999. Referred to on p.: 78

[24] P.-E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980. Referred to on p.: 75, 76, 77

[25] P.-E. Danielsson and Q. Lin. A modified fast marching method. In J. Bigün and T. Gustavsson, editors, *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, LNCS 2749, pages 1154–1161, Berlin, June 29 -July 2 2003. Springer-Verlag. Referred to on p.: 103

[26] P.-E. Danielsson, Q. Lin, and Q.-Z. Ye. Efficient detection of second-degree variations in 2D and 3D images. *Journal of Visual Communication and Image Representation*, 12:255–305, 2001. Referred to on p.: 30, 41

[27] T. Deschamps and L.D. Cohen. Fast extraction of minimal paths in 3d images and applications to virtual endoscopy. *Medical Image Analysis*, 5(4):281–299, 2001. Referred to on p.: 95

[28] R. E. Dickerson and T. K. Chiu. Helix bending as a factor in protein/dna recognition. *Biopolymers*, 44(4):361–403, 1997. Referred to on p.: 124

[29] J. Dijk, M. van Ginkel, R.J. van Asselt, L.J. van Vliet, and P.W. Verbeek. A new sharpness measure based on gaussian lines and edges. In N. Petkov and M.A. Westenberg, editors, *CAIP'03, 10th International Conference on Computer Analysis of Images and Patterns (Groningen, The Netherlands)*, volume 2756 of *Lecture Notes in Computer Science*, pages 149–156, Berlin, Aug 25-27 2003. Springer. Referred to on p.: 41, 53

[30] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. Referred to on p.: 13, 75, 79, 80

[31] L. Dorst. A coordinate system for the crystal ball. Technical report, CERN, 1980. Referred to on p.: 22

[32] D. Dunn, W.E. Higgins, and J. Wakeley. Texture segmentation using 2-d gabor elementary functions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(2):130–149, 1994. Referred to on p.: 9

[33] H. Eggers. Two fast euclidean distance transformations in z2 based on sufficient propagation. *Comput. Vision Image Understanding*, 69(1):106–116, 1998. Referred to on p.: 78

[34] F. G. A. Faas and L. J. van Vliet. A crossing detector based on the structure tensor. In J. Blanc-Talon, W. Philips, D.C. Popescu, and P. Scheunders, editors, *ACIVS 2007, Advanced Concepts for Intelligent Vision Systems, 9th International*

Conference, (Delft, The Netherlands), LNCS 4678, pages 212–220, August 28-31 2007. Referred to on p.: 27, 68

[35] F.G.A. Faas and L.J. van Vliet. 3d-orientation space; filters and sampling. In J. Bigün and T. Gustavsson, editors, *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, LNCS 2749, pages 36–42. Springer, June 29 -July 2 2003. Referred to on p.: 19, 37

[36] F.G.A. Faas, B. Rieger, L.J. van Vliet, and D.I. Cherny. Dna deformations near charged surfaces: Electron and atomic force microscopy views. *Biophysical Journal*, 97(8):1148–1157, 2009. Referred to on p.: 111

[37] Y. Fang, T.S. Spisz, and J.H. Hoh. Ethanol-induced structural transitions of DNA on mica. *Nucleic Acid Research*, 27(8):1943–1949, 1999. Referred to on p.: 112, 123

[38] G. Farnebäck. *Polynomial Expansion for Orientation and Motion Estimation*. PhD thesis, Linköping University, Linköping, Sweden, 2002. Referred to on p.: 8

[39] Faas F.G.A. and L.J. van Vliet. Junction detection and multi-orientation analysis using streamlines. In W. G. Kropatsch, M. Kampel, and A. Hanbury, editors, *CAIP 2007, Computer Analysis of Images and Patterns, 12th International Conference, (Vienna, Austria)*, pages 718–725, August 27-29 2007. Referred to on p.: 37

[40] D.J. Fleet and A.D. Jepson. Hierarchical construction of orientation and velocity selective filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(3):315–325, 1989. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/34.21800. Referred to on p.: 9, 11

[41] L.M.J. Florack, B.M. ter Haar Romeny, J.J. Koenderink, and M.A. Viergever. Scale and the differential structure of images. *Image and Vision Computing*, 10 (6):376–388, July 1992. Referred to on p.: 11

[42] W. Förstner. A feature based correspondence algorithm for image matching. *International Archives of Photogrammetry and Remote Sensing*, 26(3/3):150–166, 1986. Referred to on p.: 27

[43] C. Fouard and M. Gedda. An objective comparison between gray weighted distance transforms and distance transforms on curved spaces. In *DGCI'06, 13th International Conference on Discrete Geometry for Computer Imagery 2006 (Szeged, Hungary)*, LNCS 4242, pages 259–270, Berlin, October 2006. Springer-Verlag. Referred to on p.: 103

[44] E.M. Franken and R. Duits. Crossing-preserving coherence-enhancing diffusion on invertible orienattion scores. *International Journal of Computer Vision*, 2009. Referred to on p.: 11

[45] E.M. Franken, R. Duits, and B.M. ter Haar Romenij. Nonlinear diffusion on the 2d euclidean motion group. *Lecture notes in computer science*, 4485:461–472, 2007. Referred to on p.: 9, 11

[46] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, September 1991. Referred to on p.: 9, 11, 20, 37, 56, 70

[47] C. Frontali. Excluded-volume effect on the bidimensional conformation of DNA molecules adsorbed to protein films. *Biopolymers*, 27(8):1329–1331, 1988. Referred to on p.: 112, 123

[48] C. Frontali, E. Dore, A. Ferrauto, E. Gratton, A. Bettini, M.R. Pozzan, and E. Valdevit. An absolute method for the determination of the persistence length of native DNA from electron micrographs. *Biopolymers*, 18(6):1353–1373, 1979. Referred to on p.: 112, 115, 123, 138

[49] D. Gabor. Theory of communication. *Journal of the Institution of Electrical Engineers*, 93, Part III(26):429–457, November 1946. Referred to on p.: 9, 11

[50] J. Garding and T. Lindeberg. Direct computation of shape cues using scale-adapted spatial derivative operators. *International Journal of Computer Vision*, 17:163–191, 1996. Referred to on p.: 27, 44

[51] M. van Ginkel. *Image Analysis using Orientation Space based on Steerable Filters*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2002. Referred to on p.: 8, 11, 19, 20, 21, 23, 46, 47, 52

[52] M. van Ginkel, P.W. Verbeek, and L.J. van Vliet. Orientation selectivity for orientation estimation. In *Proceedings of the 10th Scandinavian Conference on Image Analysis (Lappeenranta, Finland)*, volume I, pages 533–537, June 9-11 1997. Referred to on p.: 37, 52, 53, 56, 70

[53] M. van Ginkel, J. van de Weijer, P.W. Verbeek, and L.J. van Vliet. Curvature estimation from orientation fields. In B.K. Ersboll and P. Johansen, editors, *SCIA'99, Proc. 11th Scandinavian Conference on Image Analysis (Kangerlussuaq, Greenland)*, pages 545–551. Pattern Recognition Society of Denmark, Lyngby, June 7-11 1999. Referred to on p.: 9

[54] D.S. Goodsell and R.E. Dickerson. Bending and curvature calculations in b-dna. *Nuclear Acid Research*, 22(24):5497–5503, 1994. Referred to on p.: 122

[55] G.H. Granlund. In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8:155–173, 1978. Referred to on p.: 11, 29, 37, 45, 55, 59

[56] G.H. Granlund and H. Knutsson. *Signal processing for computer vision*. Kluwer Academic Publishers, Boston/Dordrecht/London, 1995. Referred to on p.: 8

[57] P. Hagerman. Flexibility of dna. *Annu. Rev. Biophys. Biophys. Chem*, 17:265–286, 1988. Referred to on p.: 112, 122, 123

[58] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the fourth Alvey Vision Conference*, pages 147–151. University of Manchester, 31 August–2 September 1988. Referred to on p.: 27, 28, 37, 45, 47, 49, 55, 56

[59] R.J.van Heekeren, F.G.A. Faas, and L.J. van Vliet. Finding the minimum-cost path without cutting corners. In B. K. Ersbøll and K. Steenstrup Pedersen, editors, *SCIA 2007, Image Analysis, 15th Scandinavian Conference, (Aalborg, Denmark)*, LNCS 4522, pages 263–272, June 10-14 2007. Referred to on p.: 101, 113, 125, 128, 130, 132

[60] Y. Hel-Or and P.C. Teo. Canonical decomposition of steerable functions. *Journal of Mathematical Imaging and Vision*, 9:83–95, 1998. Referred to on p.: 11

[61] M. Herberthson, A. Brun, and H. Knutsson. Representing pairs of orientations in the plane. In B.K. Ersboll and K.S. Pedersen, editors, *SCIA'07, Proceedings of the 15th Scandinavian Conference on Image Analysis (Aalborg), Denmark)*, LNCS 4522, pages 661–670, Berlin, 2007. Springer-Verlag. Referred to on p.: 10

[62] B. Jähne. *Digital Image Processing*. Springer, Berlin, 4th edition, 1997. Referred to on p.: 8

[63] M. Joanicot and B. Revet. DNA conformational studies from electron microscopy. I. Excluded volume effect and structure dimensionality. *Biopolymers*, 26(2):315–326, 1987. Referred to on p.: 15, 112, 122

[64] P.P. Jonker. Skeletons in N dimensions using shape primitives. *Pattern Recognition Letters*, 23(6):677–686, 2002. Referred to on p.: 13

[65] N. Kaji, M. Ueda, and Y. Baba. Direct measurement of conformational changes on dna molecule intercalating with a fluorescence dye in an electrophoretic buffer solution by means of atomic force microscopy. *Electrophoresis*, 22:3357–3364, 2001. Referred to on p.: 121

[66] S.N. Kalitzin, B. ter Haar Romeny, and M.A. Viergever. Invertible apertured orientation filters in image analysis. *International Journal of Computer Vision*, 31 (2/3):145–158, 1999. Referred to on p.: 11

[67] M. Kass and A. Witkin. Analyzing oriented patterns. *Computer Vision, Graphics and Image Processing*, 37:362–385, 1987. Referred to on p.: 10, 45, 55

[68] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, pages 321–331, 1988. Referred to on p.: 13

[69] G.M.P. van Kempen, N. van den Brink, L.J. van Vliet, M. van Ginkel, and P.W. Verbeek. The application of a local dimensionality estimator to the analysis of 3d microscopic network structures. In *SCIA'99, Proc. 11th Scandinavian Conference on Image Analysis (Kangerlussuaq, Greenland)*, pages 447–455, June 7-11 1999. Referred to on p.: 46, 47, 57

[70] C. S. Kenney, M. Zuliani, and B. S. Manjunath. An axiomatic approach to corner detection. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 191–197, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2372-2. Referred to on p.: 27

[71] D. Kincaid. *Numerical analysis; mathematics of scientific computing*. Pacific Grove, Brooks/Cole, 1991. Referred to on p.: 90

[72] H. Knutsson. Representing local structure using tensors. In *The 6th Scandinavian Conference in Image Analysis*, pages 244–251, Oulu, Finland, June 19-22 1989. Referred to on p.: 19, 39, 48, 57

[73] H. Knutsson. Producing a continuous and distance preserving 5-d vector representation of 3-d orientation. In *IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management*, pages 175–182. Miami Beach, Florida, November 18-20 1985. Referred to on p.: 20, 48, 59

[74] H. Knutsson and G. H. Granlund. Fourier domain design of line and edge detectors. In *Proceedings of the 5th International Conference on Pattern Recognition*, Miami, Florida, December 1980. Referred to on p.: 20

[75] H. Knutsson and C.-F. Westin. Normalized convolution - a technique for filtering incomplete and uncertain data. In K. A. Høgda, B. Braathen, and K. Heia, editors, *SCIA'93, Proceedings of the 8th Scandinavian Conference on Image Analysis*, volume 2, pages 997–1006, Tromsø Norway, 1993. Norwegian Society for Image Processing and Pattern Recognition. Referred to on p.: 86

[76] O. Kratky and G. Porod. Röntgenuntersuchungen gelöster Fadenmoleküle. *Rec. Trav. Chim. Pays-Bas Belg*, 68:1106–1124, 1949. Referred to on p.: 14, 15, 112, 115

[77] L.D. Landau and E.M. Lifshitz. *Statistical Physics, Part 1*. Pergamon Press, second edition, 1969. Referred to on p.: 14, 112, 115, 135

[78] G. Levi and U. Montanari. A grey-weighted skeleton. *Information and Control*, 17(1):62–91, August 1970. Referred to on p.: 75, 78

[79] Q. Lin. *Enhancement, Extraction, and Visualization of 3D Volume Data*. PhD thesis, Linköping University, Sweden, 2003. Referred to on p.: 13, 14, 82

[80] Y. Lu, B. Weers, and N.C. Stellwagen. Dna persistence length revisisted. *Biopolymers*, 61(4):261–275, 2001. Referred to on p.: 112, 122

[81] C.L. Luengo Hendriks, B. Rieger, M. van Ginkel, G.M.P. van Kempen, and L.J. van Vliet. DIPimage: a scientific image processing toolbox for MATLAB, 1999-. Delft University of Technology, `http://www.qi.tnw.tudelft.nl/DIPlib`. Referred to on p.: 128

[82] S.G. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7):710–732, June 1992. Referred to on p.: 53

[83] G. S. Manning. The persistence length of dna is reached from the persistence length of its null isomer through an internal electrostatic stretching force. *Biophysical Journal*, 91(10):3607–16, 2006. Referred to on p.: 112, 122, 123

[84] J.B. Martens. Local orientation analysis in images by means of the Hermite transform. *IEEE Transactions on Image Processing*, 6-8:1103–1116, 1997. Referred to on p.: 11

[85] R.S. Mathew-Fenn, R. Das, and P.A.B. Harbury. Remeasuring the double helix. *Science*, 322:446–449, 2008. Referred to on p.: 112, 123

[86] E. Meijering. A chronology of interpolation: From ancient astronomy to modern signal and image processing. *Proceedings of the IEEE*, 90(3):319–342, March 2002. Referred to on p.: 97

[87] E. Meijering, M. Jacob, J.-C. F. Sarria, P. Steiner, H. Hirling, and M. Unser. Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images. *Cytometry*, 58A(2):167–176, 2004. Referred to on p.: 102

[88] M. Michaelis and G. Sommer. Junction classification by multiple orientation detection. In Jan-Olof Eklundh, editor, *ECCV'94, Third European Conference on Computer Vision*, pages 101–108, Stockholm, Sweden, May 1994. Referred to on p.: 11

[89] Franck Michelet, Jean-Pierre Da Costa, Olivier Lavialle, Yannick Berthoumieu, Pierre Baylou, and Christian Germain. Estimating local multiple orientations. *Signal Process.*, 87(7):1655–1669, 2007. ISSN 0165-1684. Referred to on p.: 11

[90] J.M. Moltz, I. Stuke, and T. Aach. Histogram-based orientation analysis for junctions. In F. Gini, editor, *European Signal Processing Conference*, pages 1–4 (CD–ROM), Firenze, September 4-8 2006. EURASIP. Referred to on p.: 11

[91] U. Montanari. A method for obtaining skeletons using a quasi-euclidean distance. *Journal of the Association for Computing Machinery*, 15(4):600–624, 1968. Referred to on p.: 74

[92] C. Mota, T. Aach, i. Stuke, and E. Barth. Estimation of multiple orientations in multi-dimensional signals. In *IEEE International Conference on Image Processing (ICIP)*, pages 2665–2668, Singapore, Oct. 24–27 2004. IEEE. Referred to on p.: 10

[93] J. Moukhtar, E. Fontaine, C. Faivre-Moskalenko, and A. Arneodo. Probing persistence in dna curvature properties with atomic force microscopy. *Physical Review Letters*, 98(17):178101, 2007. Referred to on p.: 15, 112, 122, 123

[94] J.C. Mullikin. The vector distance transform in two and three dimensions. *Graphical Models and Image Processing*, 54(6):526–535, 1992. Referred to on p.: 75, 78

[95] J. van Noort, S. Verbrugge, N. Gossen, C. Dekker, and R.T. Dame. Dual architectural roles of HU: formation of flexible hinges and rigid filaments. *Proceedings of the National Academy of Sciences of the United States of America*, 101(18): 6969–6974, 2005. Referred to on p.: 112

[96] A. Noy, J. Luque, and M. Orozco. Theoretical analysis of antisense duplexes: Determinants of the rnase h susceptibility. *Journal of the American Chemical Society*, 130(11):3486 –3496, 2008. Referred to on p.: 124

[97] D. Pastre, O. Pietrement, S. Fusil, F. Landousy, J. Jeusset, M. O. David, L. Hamon, E. Le Cam, and A. Zozime. Adsorption of dna to mica mediated by divalent counterions: a theoretical and experimental study. *Biophysical Journal*, 85(4):2507–18, 2003. Referred to on p.: 112

[98] A. Perez, J. Luque, and M. Orozco. Dynamics of b-dna on the microsecond time scale. *Journal of the American Chemical Society*, 129(49):14739 –14745, 2007. Referred to on p.: 124

[99] P. Perona. Steerable-scalable kernels for edge detection and junction analysis. In *ECCV'95*, pages 1–16, 1992. Referred to on p.: 11

[100] J. Piper and E. Granum. Computing distance transformations in convex and non-convex domains. *Pattern Recognition*, 20(6):599–615, 1986. Referred to on p.: 78

[101] A. Podesta, M Indrieri, D. Brogioli, G.S. Manning, P. Malani, R. Guerra, L. Finzi, and D. Dunlap. Positively charged surfaces increase the flexibility of DNA. *Biophysical Journal*, 89:2558–2563, 2005. Referred to on p.: 122, 123

[102] Y. O. Popov and A. V. Tkachenko. Effects of kinks on dna elasticity. *Physical Review E*, 71, 2005. Referred to on p.: 112

[103] G. Porod. X-ray and light scattering by chain molecules in solution. *Journal of Polymer Science*, 10(2):157–166, 1953. Referred to on p.: 115

[104] Tobias Preusser and Martin Rumpf. A level set method for anisotropic geometric diffusion in 3d image processing. *SIAM Journal on Applied Mathematics*, 62(5):p1772 –, 2002. ISSN 00361399. Referred to on p.: 9

[105] J. Radon. Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Berichte Sächsische Akademie der Wissenschaften, Leipzig, Mathematisch-Physikalische Klasse*, 69:262–277, 1917. Referred to on p.: 9

[106] I.R. Ragnemalm. Neighborhoods for distance transformation using ordered propagation. *CVGIP: Image Understanding*, 56(3):399–409, November 1992. Referred to on p.: 77, 78

[107] T.W. Ridler and S. Calvard. Picture thresholding using an iterative selection method. *IEEE Transactions on Systems, Man and Cybernetics.*, 8:630–632, 1978. Referred to on p.: 41, 64

[108] B. Rieger and L.J. van Vliet. Curvature of n-dimensional space curves in grey-value images. *IEEE Transactions on Image Processing*, 11(7):738–745, 2002. Referred to on p.: 9

[109] B. Rieger and L.J. van Vliet. Representing orientation in n-dimensional spaces. In N. Petkov and M.A. Westenberg, editors, *CAIP'03, 10th International Conference on Computer Analysis of Images and Patterns (Groningen, The Netherlands)*, LNCS 2756, pages 17–24, Berlin, August 25-27 2003. Springer Verlag. Referred to on p.: 59, 62

[110] B. Rieger and L.J. van Vliet. A systematic approach to nd orientation representation. *Image and Vision Computing*, 22(6):453–459, 2004. Referred to on p.: 48

[111] B. Rieger, F.J. Timmermans, L.J. van Vliet, and P.W. Verbeek. On curvature estimation of iso-surfaces in 3d gray-value images and the computation of shape descriptors. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 26(8):1088–1094, 2004. Referred to on p.: 9

[112] C. Rivetti, M. Guthold, and C. Bustamante. Scanning force microscopy of DNA deposited onto mica: Equilibration versus kinetic trapping studied by statistical polymer chain analysis. *Journal of Molecular Biology*, 264:919–932, 1996. Referred to on p.: 112, 115, 121, 122, 123, 128, 132

[113] C. Rivetti, C. Walker, and C. Bustamante. Polymer chain statistics and conformational analysis of DNA molecules with bends or sections of different flexibility. *Journal of Molecular Biology*, 280(1):41–59, 1998. Referred to on p.: 112, 132

[114] K. Rohr. On 3d differential operators for detecting point landmarks. *Image and Vision Computing*, 15:219–233, 1997. Referred to on p.: 27

[115] A. Rosenfeld and A.C. Kak. *Digital Picture Processing Vol. 2*. Academic Press, Orlando, 1982. Referred to on p.: 76

[116] A. Rosenfeld and J.L. Pfaltz. Distance functions on digital pictures. *Pattern Recognition*, 1(1):33–61, july 1968. Referred to on p.: 74, 78

[117] D. Rutovitz. Datastructures for operations on digital images. In G.C. Cheng, R.S. Ledley, D.K. Pollock, and A. Rosenfeld, editors, *Pictorial Pattern Recognition*, pages 105–133, Washington, DC, 1968. Thompson. Referred to on p.: 13, 75, 78

[118] D. Rutovitz. Expanding picture components to natural density boundaries by propagation methods. the notions of fall-set and fall-distance. In *Proceedings of the 4th International Joint Conference on Pattern Recognition*, pages 657–664, Kyoto, Japan, 1978. IAPR. Referred to on p.: 75

[119] E.B. Saff and A.B.J. Kuijlaars. Distributing many points on a sphere. *the Mathematical Intelligencer*, 19(1):5–11, 1997. Referred to on p.: 22

[120] P.K. Saha, F.W. Wehrli, and B.R. Gomberg. Fuzzy distance transform: Theory, algorithms, and applications. *Computer Vision and Image Understanding*, 86(3): 171–190, 2002. Referred to on p.: 103

[121] T. Saito and J. Toriwaki. New algorithms for euclidean distance transformation of an n-dimensional digitized picture with applications. *Pattern Recognition*, 27 (11):1551–1565, 1994. Referred to on p.: 78

[122] J.A. Schellman. Flexibility of DNA. *Biopolymers*, 13(1):217–226, 1974. Referred to on p.: 15, 112, 115, 138

[123] J.A. Schellman. The flexibility of DNA : I. Thermal fluctuations. *Biophysical Chemistry*, 11(3):321–328, 1980. Referred to on p.: 112, 115

[124] J.A. Schellman and S.C. Harvey. Static contributions to the persistence length of DNA and dynamic contributions to DNA curvature. *Biophysical Chemistry*, 55(1):95–114, 1995. Referred to on p.: 112, 119, 122, 138

[125] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, London, 1982. Referred to on p.: 13

[126] J. A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press, Cambridge, 1996. Referred to on p.: 13, 14, 75

[127] J.A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences of the United States of America*, 93(4):1591–1595, February 1996. Referred to on p.: 79, 82

[128] J. Shi and C. Tomasi. Good features to track. In *1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 593–600, 21-23 June 1994. Referred to on p.: 27

[129] Y. Shi, S. He, and J.E. Hearst. Statistical mechanics of the extensible and shearable elastic rod and of DNA. *The Journal of Chemical Physics*, 105(2):714–731, 1996. Referred to on p.: 115

[130] M. Shizawa and T. Iso. Direct representation and detection of multi-scale, multi-orientation fields using local differentiation filters. In *CVPR'93*, 1993. Referred to on p.: 10

[131] P. Soille. *Morphological Image Analysis. Principles and Applications*. Springer-Verlag, 1999. Referred to on p.: 13

[132] P. Thévenaz, T. Blu, and Unser M. Image interpolation and resampling. In I. Bankman, editor, *Handbook of Medical Image Processing*. Academic Press, 2000. Referred to on p.: 97

[133] P.J. Toivanen. New geodesic distance transforms for gray-scale images. *Pattern Recognition Letters*, 17(5):437–450, 1996. Referred to on p.: 75, 103

[134] E.N. Trifonov, R.K-Z. Tan, and S. C. Harvey. Static persistence length of dna. In W.K. Olson, editor, *DNA bending and curvature, Structure and Expression*, volume 3, pages 243–253. Adenine Press, Schenectady, NY, 1988. Referred to on p.: 119

[135] B. Triggs. Detecting keypoints with stable position, orientation, and scale under illumination changes. In *Eighth European Conference on Computer Vision (ECCV 2004)*, LNCS, pages 100–113. Springer-Verlag, may 2004. Referred to on p.: 27

[136] J.N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 40(9):1528–1538, September 1995. Referred to on p.: 13, 14, 75, 79, 82, 103

[137] M. Unser. Splines: A perfect fit for signal and image processing. *IEEE Signal Processing Magazine*, pages 22–38, November 1999. Referred to on p.: 97

[138] F. Valle, M. Favre, P. de los Rios, A. Rosa, and G. Dietler. Scaling exponents and probability distributions of DNA end-to-end distance. *Physical Review Letter*, 95:158105, 2005. Referred to on p.: 15, 112, 122

[139] P.W. Verbeek and B.J.H. Verwer. Shading from shape, the eikonal equation solved by grey-weighted distance transform. *Pattern Recognition Letters*, 11: 681–690, 1990. Referred to on p.: 13, 75, 103

[140] P.W. Verbeek and L.J. van Vliet. On the location error of curved edges in low-pass filtered 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(7):726–733, 1994. Referred to on p.: 58

[141] B.J.H. Verwer. Local distances for distance transformations in two and three dimensions. *Pattern Recognition Letters*, 12:671–682, November 1991. Referred to on p.: 13, 77

[142] B.J.H. Verwer, P.W. Verbeek, and S.T. Dekker. An efficient uniform cost algorithm applied to distance transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4):425–429, April 1989. Referred to on p.: 75, 77

[143] B.J.H. Verwer, L.J. van Vliet, and P.W. Verbeek. Binary and grey-value skeletons: Metrics and algorithms. *Int. Journal of Pattern Recognition and Artificial Intelligence*, 7(5):1287–1308, 1993. Referred to on p.: 13

[144] L. Vincent. Exact euclidean distance function by chain propagation. In *Proc. IEEE Computer Vision and Pattern Recognition '91*, pages 520–525, June 1991. Referred to on p.: 78

[145] L.J. van Vliet. *Grey-Scale Measurements in Multi-Dimensional Digitized Images*. PhD thesis, Delft University of Technology, Delft, The Netherlands, October 1993. Referred to on p.: 21, 90

[146] L.J. van Vliet and F.G.A Faas. Multi-orientation analysis by decomposing the structure tensor and clustering. In *ICPR18, Proceedings 18th International Conference on Pattern Recognition*, pages 856–860, Los Alamitos, Hong Kong, August 21-24 2006. IEEE Computer Society Press. Referred to on p.: 45

[147] L.J. van Vliet and P.W. Verbeek. Better geometric measurements based on photometric information. In *Proc. IEEE Instrumentation and Measurement Technology Conf. IMTC94*, pages 1357–1360, 1994. Referred to on p.: 52

[148] L.J. van Vliet and P.W. Verbeek. Segmentation of overlapping objects. In J. van Katwijk, J.J. Gerbrands, M.R. van Steen, and J.F.M. Tonino, editors, *ASCI'95, Abstracts of the ASCI Imaging Workshop*, Heijen, The Netherlands, June 1995. Referred to on p.: 11

[149] L.J. van Vliet and P.W. Verbeek. Curvature and bending energy in digitized 2d and 3d images. In *8th Scandinavian Conference on Image Analysis, Tromsø Norway*, pages 1403–1410, 1993. Referred to on p.: 58

[150] L.J. van Vliet and B.J.H. Verwer. A contour processing method for fast binary neighbourhood operations. *Pattern Recognition Letters*, 7(1):27–36, 1988. Referred to on p.: 128

[151] A.M. Vossepoel and A.W.M. Smeulders. Vector code probability and metrication error in the representation of straight lines of finite length. *Computer Graphics and Image Processing*, 20:347–364, 1982. Referred to on p.: 90

[152] A.M. Vossepoel, A.W.M Smeulders, and K. van den Broek. Dioda: Delineation and feature extraction of microscopical objects. *Computer Programs in Biomedicine*, 10:231–244, 1979. Referred to on p.: 63, 75

[153] D. Walters. Selection of image primitives for general-purpose visual processing. *Computer Vision, Graphics, and Image Processing*, 37:261–298, 1987. Referred to on p.: 11, 19

[154] J. Weickert, B.M. ter Haar Romeny, and M.A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3):398–410, 1998. Referred to on p.: 9, 128

[155] C.F. Westin. *A Tensor Framework for Multidimensional Signal Processing*. PhD thesis, Linköping University, Linköping, Sweden, 1994. Referred to on p.: 86

[156] P. A. Wiggins, R. Phillips, and P. C. Nelson. Exact theory of kinkable elastic polymers. *Physical Review E*, 71, 2005. Referred to on p.: 112

[157] P.A. Wiggins and P.C. Nelson. Generalized theory of semiflexible polymers. *Physical Review E*, 73, 2006. Referred to on p.: 112

[158] P.A. Wiggins, T. van der Heijden, F. Moreno-Herrero, A. Spakowitz, R. Phillips, J. Windom, C. Dekker, and P.C. Nelson. High flexibility of dna on short length scales probed by atomic force microscopy. *Nature Nanotechnology*, 2006. Referred to on p.: 112, 120, 123

[159] C. van Wijk, R. Truyen, R.E. van Gelder, L.J. van Vliet, and F.M. Vos. On normalized convolution to measure curvature features for automatic polyp detection. In C. Barillot, D.R. Haynor, and P. Hellier, editors, *LNCS 3216, MICCAI'04*, pages 200–208, 2004. Referred to on p.: 86

[160] W. Yu, G. Sommer, and K. Daniilidis. Three dimensional orientation signatures with conic kernel filtering for multiple motion analysis. *Image and Vision Computing*, 21(5):447–458, 2003. Referred to on p.: 11

[161] S. di Zenzo. A note on the gradient of a multi-image. *Computer Vision, Graphics and Image Processing*, 33:116–125, 1986. Referred to on p.: 10

[162] J. Zheng, Z. Li, A. Wu, and H. Zhou. Afm studies of dna structure on mica in the presence of alkaline earth metal ions. *Biophysical Chemistry*, 104:37–43, 2003. Referred to on p.: 112, 123