Technische Universität München

Zentrum Mathematik

Complexity Penalized Segmentations in 2D

Efficient Algorithms and Approximation Properties

Felix Friedrich

Vollständiger Abdruck der von der Fakultät für Mathematik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr. Dr. J. Richter-Gebert
Prüfer der Dissertation:

1. Univ.-Prof. Dr. R. Lasser
2. apl.-Prof. Dr. G. Winkler, Ludwig-Maximilians-Universität München
3. Prof. Dr. R. G. Baraniuk, Rice University Houston, Texas, USA (schriftliche Beurteilung)

Die Dissertation wurde am 30.12.2004 bei der Technischen Universität München eingereicht und durch die Fakultät für Mathematik am 18.07.2005 angenommen.

Preface

This thesis was carried out at the Institute of Biomathematics and Biometry of the GSF - National Research Center for Environment and Health, Neuherberg. It was developed in the group 'Mathematical Modelling in Ecology and the Biosciences' led by Prof. Dr. Gerhard Winkler. Without his help and encouragement, this work would not have been possible. I have learned a lot from him, not only about mathematics.

I thank Prof. Dr. Rainer Dahlhaus for motivating me to do this thesis, for mathematical impulses and for establishing the contact to Munich, and Prof. Dr. R. Lasser for his liberal supervision of this thesis.

Further, I thank the members of the GSF-Institute of Biomathematics and Biometry for their support. I am especially indebted to Hartmut Führ and Volkmar Liebscher for mathematical inspiration, sedulous help, their encouragement and patience. I am also obliged to Laurent Demaret and Olaf Wittich for long and fruitful discussions and Alan Freed for English language proof reading.

Many thanks to my friends and my family.

Munich, December 2004 Felix Friedrich

Contents

Introduction

This thesis deals with variational approaches to the segmentation of images, with a particular emphasis on efficient geometry-based algorithms.

More specifically, we consider the problem of computing segmentations of an image $f : S \to \mathbb{R}$ by minimizing certain functionals. Let us first introduce the notion of *segmentation*. A segmentation consists of a partition $\mathcal{P}$ of the image domain into connected regions, and for each such region $R \in \mathcal{P}$ a function $f_R : R \to \mathbb{R}$ from a prescribed set of functions, the *local regression model*. Piecing the functions f_R together, we obtain an approximation of f, which is denoted by $f_{\mathcal{P}}$. For the sake of concreteness, let us mention constant functions as a possible choice for the local regression model. In this setting, we are considering the problem of first partitioning the image domain into connected regions and then approximating the image by a constant on each single region.

Criteria for a choice of segmentation are usually given in the form of a variational problem, by minimization of a suitable functional

$$H(f, (\mathcal{P}, f_{\mathcal{P}})) = U(\mathcal{P}, f_{\mathcal{P}}) + D(f, (\mathcal{P}, f_{\mathcal{P}}))$$

in $(\mathcal{P}, f_{\mathcal{P}})$, ranging over a certain prescribed *segmentation class*. The term D measures the distance of segmentations $(\mathcal{P}, f_{\mathcal{P}})$ to image data f, and U is a penalty or regularization term. There are various motives for considering such a problem. One could be *denoising*: Here we assume that the image f originated from an element of the segmentation class by some distortion or noise effect and we want to recover the original image. In this setting, U codes prior information on the original, whereas D incorporates knowledge about the noise. Another application could be *compression*: Here, U is the coding cost for storing the segmentation, and D is a measure of distortion arising from compression. Finally, segmentations contain information about the image, which makes them useful tools for *image analysis* purposes. For instance the constant model is clearly related to the problem of edge detection.

While such functionals provide an elegant method to single out particularly useful segmentations, the arising minimization problems very often turn out

to be intractable. The predominant approaches can be grouped roughly as follows:

- Stochastic optimization methods like simulated annealing or other Metropolis-Hastings type algorithms, see Gilks et al. (1996), Winkler (2002), Casella and Robert (1999). Such methods never give exact results, since they spend a lot of simulation time in local minima and do not provide decision rules to decide if they are already in a global minimum; even quality estimates by convergence diagnostics or related techniques – albeit frequently used in practice – are questionable.
- Deterministic minimization in a restricted search space. The motivation behind this approach is the desire to obtain efficient algorithms that solve the restricted problem in an exact manner. In this setting we are in a conflict between feasibility of algorithms and sufficient richness of the search space. A further criterion for the design of the search space is given by the desire to mathematically analyze the results of the minimization procedure.

In this thesis we subscribe to the second approach. More precisely, we propose two classes of partitions, whose structures permit fast minimization algorithms. *Hierarchic partitions* are formed by subsequent division of the image domain in the horizontal and vertical directions. By their recursive structure, they admit a minimization algorithm that consists of an iterative application of a dynamic programming approach in one dimension, see Winkler and Liebscher (2002). *Dyadic partitions* are generated by recursively applying a quad-split to the image domain. The tree structure induced this way will be utilized for the development of a minimization algorithm. Both dyadic and hierarchic partitions consist of rectangles. By additionally splitting the rectangles along straight lines, we arrive at the set of *wedges*, designed to better approximate the geometric fine structure of the image. We control the angular resolution of the arising partitions by a priori fixing the set of lines. The resulting partitions will be called wedge- or wedgelet-partitions.

For local regression models, we use the restrictions of elements of a fixed finite dimensional function space over the image domain S to the respective region. This includes as special case the locally constant model.

The functional under consideration will be

$$H_\gamma : (f, (\mathcal{P}, f_\mathcal{P})) \longmapsto \gamma \cdot |\mathcal{P}| + \|f - f_\mathcal{P}\|_2^2,\ \gamma \geq 0.$$

In this thesis, functionals of this form will be called *Potts functionals* with parameter γ. We chose this name since a similar penalty was first used in Potts (1952) as the energy function of a Gibbsian model in statistical mechanics, generalizing the Ising model (from Ising (1925)) from binary random fields to those with a finite number of spins. The parameter γ weighing the distance versus the penalty term can be interpreted as inverse smoothness or scale parameter. For $\gamma = 0$ the minimizer will in general be given by the finest possible segmentation, whereas for $\gamma \to \infty$ the segmentation will consist of a single region. As a consequence of taking the squared L^2-norm for the distance term, we obtain that a segmentation $(\mathcal{P}, f_{\mathcal{P}})$ is uniquely described by the partition $\mathcal{P}$; $f_{\mathcal{P}}$ is obtained from $\mathcal{P}$ by orthogonally projecting f onto suitable function spaces. In the following we call the computation of these projections the *local regression problem*.

In the one-dimensional case the Potts functional and its minimizers have been thoroughly investigated, e.g. in Kempe (2003), and Winkler and Liebscher (2002). In particular, dynamic programming techniques have been derived and implemented, that allow the rapid computation of exact minimizers without any restriction on the search space. These techniques are the basis for our treatment of the hierarchic model. Observe that there is a one-to-one and onto correspondence between one-dimensional signals f and locally constant segmentations $(\mathcal{P}, f_{\mathcal{P}})$ (provided that the segmentations are minimal in the sense that no two adjacent intervals carry the same value). In higher dimensions, the number of jumps in a signal and the cardinality of partitions are no more related in the simple way as they are in one dimension. The increase in complexity in the two-dimensional case necessitates restrictions on the search space like the ones we propose in this thesis.

A special case of the dyadic partition scheme are the wedgelets originally proposed by Donoho (1999), designed to overcome certain shortcomings of two-dimensional wavelets. It is well-known that nonlinear approximation, with respect to a wavelet orthonormal basis, achieved by a simple sorting and truncation of coefficients, yields optimal approximation rates for one-dimensional piecewise smooth signals. The deeper reason is that the vanishing moments of the wavelets imply a fast decay of wavelet coefficients away from the singularities. In view of the wide success of wavelets, in particular in the domain of image compression, it is somewhat surprising that this approximation behavior does not pertain in the two-dimensional case: The usual tensor product construction of a two-dimensional multiresolution analysis results in a wavelet orthonormal basis that has – provably– subopti-

mal nonlinear approximation rates, see e.g. the discussion in Mallat (1998). This fact motivated various researchers to develop alternative systems and approximation schemes. In particular Donoho is responsible for the creation of two main branches of this development, namely curvelets [Starck et al. (2000)] – and their relatives, called ridgelets [Candès (1999); Candès and Donoho (1999)] and contourlets [Do and Vetterli (to appear.)] –, as well as wedgelets (which spawned, amongst other constructions, platelets, compare Willett and Nowak (2003)). Roughly speaking, the first group is based on methods from harmonic analysis, sometimes using group-theoretic language. These tools are employed for the construction of a suitable system of building blocks, usually forming a tight frame of the image space, and approximation is achieved by truncating the expansion of an image with respect to this frame. By contrast, the second group incorporates geometric image domain partitioning with local regression. Intermediate constructions were also developed, such as the bandelets due to Pennec and Mallat (to appear), which are based on a combination of partitioning and (warped) wavelet analysis.

Already in the initial paper [Donoho (1999)], Donoho noted the need for the fast computation of local regressions, which in the wedgelet case amounts to computing mean values over wedge-shaped domains of varying shapes and sizes. His algorithms were based on the assumption that these values were computable in reasonable time, but judging from the literature and the few existing freely available implementations available at Donoho et al. (2004) and Willet (2004), no solution to this problem has so far been presented. Some indication was given in the paper Romberg et al. (2002), which used two-scale relations between wedges of different scales, the idea being that a wedge of size 2^j could be pieced together from wedges of size 2^{j-1}. However, the exposition is far from complete and is hardly a base for an efficient implementation. Greedy algorithms for wedgelet type approximation were developed by Willett et al. (2004). In terms of computing times these routines come near our algorithms, at the cost of yielding only local minima of the cost function. Besides being undesirable for image processing purposes, this suboptimal behavior also poses severe problems for the mathematical analysis of the algorithm.

The main contribution of this thesis can be summarized by the following two points:

- The design of fast and flexible algorithms for the computation of wedgelet-type approximations, based on a particularly efficient solution of a local regression problem. Here, the angular resolution can be prescribed in

an arbitrary manner. Moreover, for dyadic partitions, the implementation gives fast access to the entire scale of minimizers.

- The formulation and proof of consistency results for the wedgelet estimators. These results describe the asymptotic behavior of wedgelet-minimizers of discretized continuous images, as the pixel size tends to zero. They provide heuristics for the choice of the parameter γ and yield convergence rates that, for certain regression models, are close to theoretically optimal.

On the algorithmic side, we present a methodology which allows the flexible design of wedgelet-like systems, along with powerful algorithms for efficient approximation in these systems. It allows to treat wedgelets as well as the platelets developed by Willett and Nowak, which use piecewise affine instead of piecewise constant images. It is one of the main achievements of this thesis to make fast algorithms for the precise computation of wedgelet approximations available. A careful treatment of digital lines turns out to be a key feature. It allows the development of rapid summation methods for polygonal domains with a prescribed angular resolution. The angular resolution can be prescribed in a direct and convenient way, allowing a more or less arbitrary control of the tradeoff between geometric accuracy and computation time.

The dramatic savings in computing time, which result from the new techniques, open new fields of applications for wedgelet approximations. It is safe to say that before this speedup, wedgelet decompositions were mainly of theoretical interest. With existing packages, it has been impossible to compute wedgelet approximations for average-sized images in reasonable time; the numerical experiments presented in Chapter 6 would have taken years with BeamLab. By contrast, it is now possible to compute the wedgelet decomposition of a 500 x 500 pixel image in less than a minute, and to investigate the influence of the regularization parameter in real time. Thus, the full range of approximations is directly accessible.

The gain in speed could be invested into more complex – and thus, more realistic – local models. Also, the fact that the whole scale of approximations, ranging from coarse (for large hyperparameters) to fine, is available at negligible extra computational cost, turns wedgelet approximation into a useful tool for image analysis purposes. Thus, wedgelet approximations can be used as a means of extracting (and processing) multiscale information contained in an image. In Section 6.4, we sketch a sample application using this feature

of our wedgelet implementation in the context of denoising.

On the theoretical side, we discuss the method of deriving wedge segmentations by minimizing Potts functionals with respect to statistical consistency and rates of convergence. Our assumptions on the noise are clearly beyond standard assumptions: we generalize them in two ways. First, independent gaussian random variables are replaced by independent subgaussian random variables. Second, we permit heterogeneity of the scale of the noise, whereas random variables are typically required to be identically distributed. For a minimizer $(\hat{\mathcal{P}}, \hat{f}_\gamma)$ of the Potts functional, we give an a priori estimate of $\|f - \hat{f}_\gamma\|_2^2$ consisting of a sum of a deterministic approximation error of the true signal and a contribution of the segmented noise. The former leads to approximation spaces, which are at the basis of nonlinear approximation theory, in a natural way. The latter, stochastic, contribution is easily controlled by maximal inequalities for subgaussian random fields. Due to this abstract formulation, this result applies to a much wider class of segmentations than the studied hierarchic and dyadic wedge segmentations. For horizon functions and smooth functions, this yields near optimal rates of convergence.

We will now give a detailed overview of this thesis.

In **Chapter 1**, we provide a detailed introduction of the notion 'segmentation' and of the associated functionals. We present the so called reduction principle that will turn out to be the key for the development of efficient algorithms. In fact it provides conditions for a generalization of the minimization algorithms developed in this thesis to other functionals than the Potts functional. We display a naive minimization procedure in the arbitrarily-dimensional case, and we comment on a dynamic programming algorithm for the minimization of Potts functionals in one dimension that will be a fundamental component in the minimization algorithm for hierarchic segmentations.

In **Chapter 2**, two partition classes are presented that permit an efficient minimization algorithm: hierarchic and dyadic partitions. Hierarchic partitions are formed by subsequent division of the image domain in the horizontal and vertical directions. Dyadic partitions are introduced by way of a restriction of their fragments to dyadic squares. We establish that they carry a quad-tree structure. In both cases we comment on the relationship between partitions over continuous and discrete image domains. Algorithms for efficient traversing of both partition classes for the purpose of minimizing Potts functionals are presented together with their complexity properties. These

algorithms are based on the recursive structure of the partition classes.

Chapter 3 deals with the refinement of hierarchic and dyadic partitions by introducing wedges. The (local) regression problem is also addressed. We start with general remarks about existence and uniqueness of minimizers of different functionals. In particular, the least squares regression problem is addressed. We show that for an efficient computation of the respective projections, a fast computation of certain moments of the image is essential. Then we introduce and characterize wedges, formed by division of rectangles by intersection with a line. We investigate digital lines for the purpose of a detailed treatment of the relationship between discrete and continuous wedges, and we characterize the set of discrete wedges by a detailed discussion of the geometric features of linear dichotomies. As a byproduct of this discussion, an algorithm for the determination of ordered pairs associated to a linear dichotomy is presented. In the third part of Chapter 3, we present a method for efficient regression over polygonal image domain. Such a method is a crucial ingredient for the implementation of efficient algorithms for a minimization of the Potts functional over wedge segmentations. We derive properties of two intersecting Jordan curves in the plane and utilize them for an integration formula over polygonal domain on the lattice. We show that an integration of a function over polygonal domains can be performed with a complexity of number of vertices, provided that certain matrices have been precomputed for each direction of the edges contained in the respective polygon. This leads us to an efficient integration algorithm for a known prescribed class of polygonal domains.

At the center of this thesis, in **Chapter 4**, we combine and exploit the results about partitions and local regression of the previous chapters, and we present an algorithm for an efficient minimization of the Potts functional for wedge segmentations. For that purpose, we introduce wedge segmentations where an angular resolution is prescribed by the specification of an arbitrary finite set of angles. We present an algorithm for the determination of the best approximating wedge division of a rectangle corresponding to a given set of angles. By using this algorithm as a 'plug-in' in the fast traversing algorithms for dyadic and hierarchic partitions, we present the algorithms for efficient minimization of the Potts functional together with their complexity estimates.

In **Chapter 5**, segmentation classes are investigated with respect to statistical consistency. We start with a set-up borrowed from nonparametric regression. There we assume that an observation is given as an erroneous,

discrete measurement of some unknown, continuous 'truth'. We derive the needed results concerning the relation between continuous and discrete signals, in particular with respect to the evaluation of Potts functionals. The requirement on the noise to be subgaussian is devised, and some consequences are derived. We present the main results of this chapter formulated as general as possible in a second part. There we present the crucial tool for controlling the effect of noise on the segmentation, a maximal inequality for subgaussian noise. We state conditions on classes of segmentations, for which the consistency results are formulated. Then we derive the main result of this chapter: We show that if a certain approximation rate is given on the continuous side and if the discretization error is bounded in a given way, then consistency and even a rate of convergence holds for a wide class of segmentations. In the third part of Chapter 5, we apply this result to dyadic and hierarchic wedge segmentations. Two scenarios from the literature, piecewise polynomial approximation of smooth functions and constant approximations of horizon functions, will also be investigated. For piecewise polynomial approximations we obtain consistency for the hierarchic wedge segmentations and additionally a rate of convergence for dyadic wedge segmentations and sufficiently smooth original image. For constant approximations, we can solve the problem of discretization and approximation separately and formulate results for horizon functions. In this special case for hierarchic segmentations we prove a convergence rate that is superior to that of dyadic segmentations.

Chapter 6 contains mainly experimental results. Features and structure of a software implementation of the provided algorithms are described. Runtime measurements support the high efficiency of both the algorithms and the implementation and experimentally verify the complexity results presented in previous chapters. We comment on suitable scaling of the parameter γ for comparability of images with different resolution and grey-value range. Typical outcomes of the algorithms for different types of image data are presented for a purely phenomenological analysis. By strict separation of the regression part of the algorithm and the minimization routine, we have provided a method to access a segmentation minimizing the Potts functional to different parameters γ more or less immediately. This gives rise to further experiments on a multiresolution scale. Plots of the Peak to Signal Noise Ratio (PSNR) versus the number of pieces provide hints for image compression and the choice of angular resolution. We provide a reasoning for an adaptive scheme for the angular resolution leading to even faster algorithms. A simple experiment with horizon functions reveals the problematic nature of asymptotic results, which is caused by discretization effects. In the last

part of this chapter, we present experiments for denoising of images. We show an interesting effect that occurs, when segmentations are displayed for increasing values of γ on a logarithmic scale. Simulations of noisy images and a noisy flat for different distributions and different variances reveal that the noise and the corresponding resolution parameter γ can be robustly identified by a certain slope in the corresponding curves displaying log number of segments against log γ.

We close this thesis with the **Discussion and Outlook** on page 221.

The most important **symbols** used in this thesis are summarized in a nomenclature on page 224.

1. Segmentations and the Reduction Principle

In this first chapter, we introduce the basic notion used in this thesis: segmentations. Then we present the reduction principle that is one of the keys for the development of efficient algorithms. At last we comment on a naive algorithm for arbitrary dimensions and on a dynamic programming approach in the one-dimensional case.

1.1 Segmentations

In this section, we introduce segmentations. Although the algorithms proposed in this thesis operate on finite data, we introduce them for both a finite (discrete) and a continuous index set. This has been done for the following reasons: First, the geometrical restrictions that are imposed on the class of admissible partitions can be formulated much easier on the continuous domain. Shapes such as triangles, curves and wedges can be naturally defined on – say, $\mathbb{R}^2$. The discretization, namely the identification of regions belonging to such shapes in some discrete domain, for example $\mathbb{Z}^2$, is then solely understood as being that process needed to make these objects accessible in the discrete domain. Second, we might assume that data originate from some continuous resource. The observed signal shall then be deemed a discretized (and possibly distorted) version of some continuous (unknown) 'truth'. In this context a typical question is: Will increasing resolution of the measurement result in better approximations of the 'truth'? We need the parallel formulation for continuous and discrete image domain as preparation for Chapter 5, where questions of this kind will be answered.

1.1.1 Partitions

A segmentation will consist of a partition of the image domain into mutually disjoint and connected regions, and the specification of a real valued function for each region. Before we define the first ingredients, namely partitions, we

need some notation:

A neighborhood system ∂ in a countable set S is a set $\partial = \{\partial(s) : s \in S\}$ of subsets of S with $s \notin \partial(s)$ and $s \in \partial(t)$ if and only if $t \in \partial(s)$ for all pairs $s \neq t$, $s, t \in S$. S is said to be endowed with the neighborhood structure ∂. A set $A \subset S$ is called connected, if for each $s, t \in A$, there is some path $(p_i)_{1 \leq i \leq n}$, $n \in \mathbb{N}$, with $p_1 = s$, $p_n = t$, $p_i \in A$ and $p_i \in \partial(p_{i+1})$ for all $1 \leq i < n$. Recall that a set B in a topological space is called connected if there are no two open disjoint subsets B_1 and B_2 of B such that $B_1 \cup B_2 = B$.

Signals are real valued functions living on an image domain S. Image domains can be either continuous, e.g. $S = [0,1)^d$ ($d \in \mathbb{N}$), or discrete, e.g. $S = \prod_{i=1}^{d} \{1, \ldots, d_i\}$ ($d_i \in \mathbb{N}$ for all $1 \leq i \leq d$, $d \in \mathbb{N}$). In the following, notions such as connectedness depend on the context. In the continuous setting we refer to the usual metric topology on $\mathbb{R}^d$, in the discrete case it is defined by a neighborhood structure.

Let us now introduce the partitions.

Definition 1.1.1 (Partition, Fragment): *Let S be an image domain and $\mathcal{R}$ a family of subsets of S. A finite set $\mathcal{P} \subset \mathcal{R}$ is called a* **partition** *of S if the following three conditions hold:*

(A1) $S = \bigcup_{p \in \mathcal{P}} p$.

(A2) $r \cap q = \emptyset$ *for all* $r, q \in \mathcal{P}$ *with* $r \neq q$.

(A3) *All* $r \in \mathcal{P}$ *are connected.*

An element r of a partition $\mathcal{P}$ will be called a **fragment***; the set $\mathcal{R}$ is called* **the set of admissible fragments**.

In the following, the family of all partitions with elements from $\mathcal{R}$ will be denoted by $\mathfrak{P} = \mathfrak{P}(\mathcal{R})$.

Remark 1.1.2: *The canonical discretization $\boldsymbol{\delta}$ that maps subsets A of the continuous domain $I = [0,1)^2$ to subsets $\boldsymbol{\delta}(A)$ of $S = \{1, \ldots, N_1\} \times \{1, \ldots, N_2\}$, given by*

$$\boldsymbol{\delta}(p) = \{(N_1 s_1 + \frac{1}{2}, N_2 s_2 + \frac{1}{2}) : (s_1, s_2) \in p\} \cap \mathbb{Z}^2,$$

applied to each element of a partition separately, does not necessarily transform a partition of I into a partition of S because in general, it does not preserve connectedness (even not necessarily for convex sets, see illustration in Figure 1.1). This is another reason why Definition 1.1.1 is formulated for both continuous and discrete image domains.

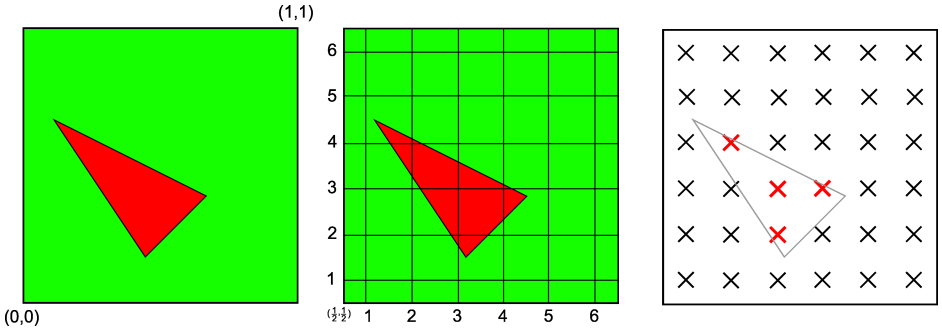


Fig. 1.1: *The discretization of a connected, convex set is not necessarily connected (here: with respect to nearest (4-)neighborhood).*

However, the partitions used in this thesis for an efficient minimization of the Potts functional are designed in such a way that the discretization $\boldsymbol{\delta}$ maps a set of partitions into a set of partitions.

Example 1.1.3: *Consider the one-dimensional continuous interval $S = [0, 1)$. Since the only connected sets of the real line are intervals, the partitions of $[0, 1)$ are given by disjoint division of $[0, 1)$ into intervals. Moreover, the discretization operation $\boldsymbol{\delta} : [0, 1) \to \{1, \ldots, N\}$ preserves connectedness since it maps intervals (a, b) to discrete intervals $(aN + \frac{1}{2}, bN + \frac{1}{2}) \cap \mathbb{Z}$. Choosing $\mathcal{R}$ to be the set of intervals in $[0, 1)$ leads to a family $\mathfrak{P}(\mathcal{F})$ consisting of all possible partitions of $[0, 1)$.*

Fig. 1.2: *Partitions of the interval $[0, 1)$, right: dyadic partition*

To mention a meaningful restriction on the class of admissible fragments (and thus the class of partitions) consider $\mathcal{R} = \{[(k-1)2^{-n}, k2^{-n}), 1 \leq k \leq 2^n, n \in \mathbb{N}\}$, the set of dyadic intervals. The set $\mathfrak{P}(\mathcal{R})$ of partitions of $[0, 1)$ are then the so called dyadic partitions. These will be generalized in Chapter 2 for two dimensions.

1.1.2 Segmentations

Now we introduce segmentations: Partitions that are additionally provided with real valued functions on each fragment. A segmentation class is given by a partition class together with function spaces for each admissible fragment. A segmentation class does not only determine the permitted geometry of the image domain, but also the admissible form of the image data.

Definition 1.1.4 (Admissible Functions, Segmentations): *Let $\mathcal{R}$ be a set of admissible fragments, as defined above. Let $\mathcal{F}$ denote a family of function spaces, $\mathcal{F} = (\mathcal{F}_p)_{p\in\mathcal{R}}$, where $\mathcal{F}_p$ is a set of functions from p to the real line $\mathbb{R}$ for each $p \in \mathcal{R}$. We will call $\mathcal{F}$ the class of* **admissible functions**. *Consider a partition $\mathcal{P} \in \mathfrak{P}(\mathcal{R})$ of S. A* **Segmentation P** *is the pair of a partition $\mathcal{P} \in \mathfrak{P}(\mathfrak{R})$ and a function vector*

$$f_{\mathcal{P}} := (f_p)_{p\in\mathcal{P}} \in \bigtimes_{r\in\mathcal{P}} \mathcal{F}_r,$$

i.e. for each p, the function f_p is admissible. We write

$$\mathbf{P} = (\mathcal{P}, f_{\mathcal{P}}).$$

A family $\mathfrak{S}$ of segmentations will be called a **segmentation class**. *An element $\mathbf{p} = (p, f_p) \in \mathbf{P}$ of a segmentation $\mathbf{P}$, that is a fragment p together with its function f_p will be called a* **segment**.

We denote a segmentation class with partitions $\mathfrak{P}$ and functions $\mathcal{F}$ by $\mathfrak{S} = \mathfrak{S}(\mathfrak{P}, \mathcal{F})$.

Usually, special notions of smoothness are captured by the function class. To exemplify that, we continue with the one-dimensional case in the following example.

Example 1.1.5: *Let $S = [0, 1)$ and $\mathfrak{P}$ be the class of all partitions over $[0, 1)$.*

- *Let $\mathcal{F}$ be the class of constant functions over fragments, $\mathcal{F}_p = \{f \in map(p, \mathbb{R}) : f(x) = \mu \ \forall x \in p, \mu \in \mathbb{R}\}$ for each $p \in \mathcal{R}$. The class of segmentations $\mathfrak{S}(\mathfrak{P}, \mathcal{F})$ can then be identified with piecewise constant functions on $[0, 1)$.*

- *Let $\mathcal{F}$ be the affine functions over fragments, $\mathcal{F}(p) = \{f \in map(p, R) : f(x) = \mu + \lambda x,\ \mu, \lambda \in \mathbb{R}\}$ for each $p \in \mathcal{R}$. The class of segmentations $\mathfrak{S}(\mathfrak{P}, \mathcal{F})$ can be identified with piecewise affine functions on $[0, 1)$.*

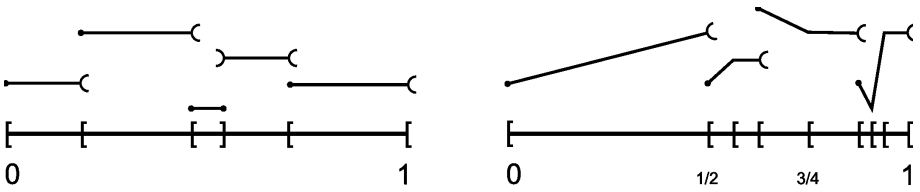


Fig. 1.3: *Piecewise constant and piecewise affine segmentation over $[0, 1)$.*

If, more generally, $\mathcal{F}$ is the class of polynomial functions over fragments, then the class of segmentations $\mathfrak{S}(\mathfrak{P}, \mathcal{F})$ can be identified with piecewise polynomial functions on $[0, 1)$.

A segmentation $\mathbf{P} = (\mathcal{P}, f_{\mathcal{P}})$ generates a function g from the index set S to the real line $\mathbb{R}$ in the following canonical way:

$$g(s) = f_p(s),\ s \in p, p \in \mathcal{P}.$$

We use the suggestive notation $g = f_{\mathcal{P}}$ and the symbol $f_{\mathcal{P}}$ for both the family of functions $(f_p)_{p \in \mathcal{P}}$ and the function g.

Now fix a segmentation class $\mathfrak{S} = \mathfrak{S}(\mathfrak{P}, \mathcal{F})$, i.e. a class of admissible partitions and functions. Assume, conversely, a real valued function g on S is given. If there is some $\mathcal{P} \in \mathfrak{P}$ such that $g|_r \in \mathcal{F}_r$ for all $r \in \mathcal{P}$, then we say that the partition $\mathcal{P}$ **$\mathcal{F}$-segments** the function g. Note that for such a fixed partition $\mathcal{P}$ the functions $f_r = g|_r \in \mathcal{F}_r$ $(r \in \mathcal{P})$ are unique. Nevertheless, neither existence nor – in case of existence – uniqueness of the so induced segmentation can be taken for granted, as the following example shows.

Example 1.1.6: *Let a set $S = \{1, 2, 3\}$ and data $z = (0, 1, 0)$ be given. Let the class of admissible functions $\mathcal{F}$ be affine functions on intervals of S. If the partition class is given by $\mathfrak{P} = \{S\}$, then since $g \notin \mathcal{F}_S$, there is no segmentation with $f_{\mathcal{P}} = g$. If the partition class is given by all admissible partitions of $\{1, 2, 3\}$, then the following solutions are possible:*

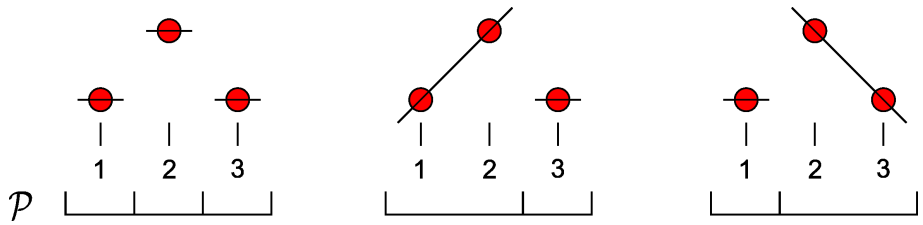


Fig. 1.4: *Three different piecewise linear segmentations for* $z = (0, 1, 0)$.

We will call a partition $\mathcal{P}$ *coarser than* $\mathcal{Q}$*, if for each* $p \in \mathcal{P}$ *there is a subset* $\mathcal{A}$ *of* $\mathcal{Q}$ *such that* $\mathcal{A}$ *is a partition of* p*. The example shows also that even the requirement of maximality of the partitions with respect to coarseness does not imply uniqueness of the segmentation fulfilling* $f_{\mathcal{P}} = g$*. The reason is that the partitions that are admitted by the classes* $\mathfrak{P}$ *and* $\mathcal{F}$ *with* $f_{\mathcal{P}} = g$ *are no lattice (although the set of all partitions* $\mathfrak{P}$ *is a lattice).*

1.2 The Reduction Principle

The reduction principle is very much at the heart of this thesis. If it applies, then the minimization of a functional over segmentations can be separated into a subsequent minimization over the two components. First, a minimizing family of functions is determined for each partition, and second, the minimum is taken over all partitions. This has two relevant consequences:

1. It is the key to the development of fast algorithms;
2. It allows to restrict minimization over partitions to subclasses of all the partitions, in cases where minimization over all partitions is computationally not feasible.

Thus, a discussion of the feasible partitions can and must be separated from a discussion of the general minimization problem. This text is mainly about the Potts functional. However, the following abstract introduction of the reduction principle shows that concepts and algorithms can be applied in quite general situations.

1.2.1 The Reduction Principle for Segmentations

As introduced in Section 1.1, the symbol $\mathcal{R}$ stands for a family of admissible fragments of the index set S, and $\mathcal{F} = (\mathcal{F}_r)_{r \in \mathcal{R}}$ is a class of real valued

functions. Recall that $\mathfrak{P} = \mathfrak{P}(\mathcal{R})$ denotes a family of partitions, and $\mathfrak{S} = \mathfrak{S}(\mathfrak{P}, \mathcal{F})$ is a family of segmentations.

Definition 1.2.1 (The Reduction Principle): *Let H be some real valued functional on the class of segmentations $\mathfrak{S}(\mathfrak{P}, \mathcal{F})$. Consider the minimization problem*

$$H(\mathcal{P}, f) \xrightarrow{!} \min_{(\mathcal{P},f)\in\mathfrak{S}} H(\mathcal{P}, f). \tag{1.1}$$

We say that the **Reduction Principle** *holds for the minimization problem (1.1), if and only if there are functions $h_r \in map(\mathcal{F}_r, \mathbb{R})$, $r \in \mathcal{R}$, and if for each $\mathcal{P} \in \mathfrak{P}$ there is a function $\varphi_{\mathcal{P}} : \mathbb{R}^{\mathcal{P}} \to \mathbb{R}$ such that $H(\mathcal{P}, f_{\mathcal{P}}) = \varphi_{\mathcal{P}}((h_r(f_r))_{r\in\mathcal{P}})$ and such that the (global) minimum of H is given with*

$$h_r^* = \min_{f_r\in\mathcal{F}_r} h_r(f_r),\ r \in \mathcal{R}$$

by

$$\min_{(\mathcal{P},f)\in\mathfrak{S}} H(\mathcal{P}, f) = \min_{\mathcal{P}\in\mathfrak{P}} \varphi_{\mathcal{P}}((h_r^*)_{r\in\mathcal{P}}).$$

The reduction principle implies that the function H contains no explicit interaction between different segments in a segmentation. The value of H on the different fragments of P is separable in the sense that minimization over $\mathfrak{S}(\mathfrak{P}, \mathcal{F})$ splits up into a minimization over $\mathcal{F}_r$ within each fragment r of any partition, and a minimization over all partitions afterwards.

Remark 1.2.2: *The reduction principle is a statement about the functional H, and not about the segmentation class $\mathfrak{S}$. By the separation of partitions and functions in this thesis, the design of a segmentation class with explicit interaction between function classes on different fragments within one partition is impossible a priori. The set of segmentations minimizing some functional H however might contain such interactions if the reduction principle does not hold for H.*

We illustrate the preceding remark and the reduction principle with the following counterexamples. They show that interactions between segments of a partition can neither be modelled with the chosen segmentation class nor – if the reduction principle applies – with the minimization of a functional H.

Example 1.2.3: *Consider the index set $S = [0,1)$, $\mathcal{R}$ the set of intervals in S and $\mathfrak{P}(\mathcal{R})$ the class of all partitions of S.*

(1.) The set $\mathcal{S} = \{(\mathcal{P}, f_\mathcal{P}) : f_r$ affine for all $r \in \mathcal{P}$, $f_\mathcal{P}$ continuous $, \mathcal{P} \in \mathfrak{P}\}$ is not a segmentation class. If it was, then defining the sets $\mathcal{F}_r = \{f_\mathcal{P}|_r : r \in \mathcal{P}, (\mathcal{P}, f_\mathcal{P}) \in \mathcal{S}\}$ $(r \in \mathcal{R})$ and $\mathfrak{S} = \{(\mathcal{P}, f_\mathcal{P}) : f_\mathcal{P} \in \prod_{r\in\mathcal{P}} \mathcal{F}_r, \mathcal{P} \in \mathfrak{P}\}$ would yield $\mathcal{S} = \mathfrak{S}$, but $\mathfrak{S}$ is strictly larger than $\mathcal{S}$.

(2.) Let now $\mathcal{F}_r$ be the family of affine functions for each interval $r \subset [0,1)$, and let the segmentation class be given by $\mathfrak{S} = \mathfrak{S}(\mathfrak{P}, \mathcal{R})$ (inducing piecewise affine functions). Consider the functional $G : \mathfrak{S} \to \mathbb{R}$ given by

$$G(\mathcal{P}, f_\mathcal{P}) = \begin{cases} 0 & \text{if } f_\mathcal{P} \text{ is continuous,} \\ 1 & \text{otherwise.} \end{cases}$$

The reduction principle does not hold for the minimization of G: Fix $\mathcal{P} = \{[0, \frac{1}{2}), [\frac{1}{2}, 1)\}$ and define functions $f_1, g_1 : [0, \frac{1}{2}) \to \mathbb{R}$ and $f_2, g_2 : [\frac{1}{2}, 1) \to \mathbb{R}$ by $f_1(x) = \frac{1}{2}x$, $f_2(x) = 1 - \frac{1}{2}x$, $g_1(x) = -\frac{1}{2}x$ and $g_2(x) = \frac{1}{2}x - 1$. Clearly the function pairs (f_1, f_2) and (g_1, g_2) induce continuous functions and are thus minimizers of G.

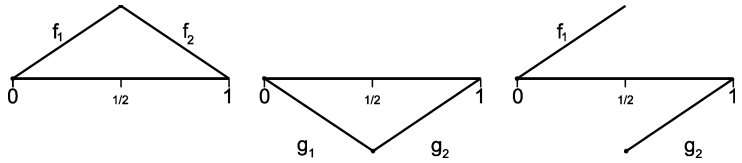


Fig. 1.5: *f_1, f_2, g_1 and g_2.*

If the reduction principle held, then there would be some function $h_r : \mathcal{F}_r \to \mathbb{R}$ such that

$$\hat{f}_\mathcal{P} \in \operatorname*{argmin}_{f_\mathcal{P} \in \mathcal{F}_\mathcal{P}} G(\mathcal{P}, f_\mathcal{P})$$

is equivalent with

$$\hat{f}_r \in \operatorname*{argmin}_{f \in \mathcal{F}_r} h_r(f_r)$$

for each $r \in \mathcal{P}$. But then the pair (f_1, g_2) would also be a minimizer of G – a contradiction.

(3.) An example of a functional where the reduction principle applies is the Potts model, which will be introduced in the next section.

Note that even if the reduction principle holds for some functional H, the minimization can still be a serious problem. On the one hand, the local minimization may itself be a complex task, on the other hand, the set of admissible partitions is, in general, very large. A carefully chosen, large enough segmentation class can be a way out of this misery. This text addresses such partition classes.

Now on $\mathbb{R}^n$ ($n \in \mathbb{N}$), let the half order $\prec$ be given by $v \prec w$ ($v, w \in \mathbb{R}^n$) if and only if $v_i < w_i$ for all $1 \leq i \leq n$. A wide class of functionals fulfilling the reduction principle consists of functionals that are of the form $\varphi_{\mathcal{P}}((h_r)_{r\in\mathcal{P}})$, where for all $\mathcal{P} \in \mathfrak{P}$ the real valued function $\varphi_{\mathcal{P}}$ is strictly monotonically increasing, i.e. $\varphi_{\mathcal{P}}(v) < \varphi_{\mathcal{P}}(w)$ if $v \prec w$ ($v, w \in \mathbb{R}^{\mathcal{P}}$).

Theorem 1.2.4: *Let a functional $H : \mathfrak{S}(\mathfrak{P}, \mathcal{R}) \to \mathbb{R}$ be given. If for all $r \in \mathcal{R}$ there are functions $h_r \in map(\mathcal{F}_r, \mathbb{R})$, and if for each $\mathcal{P} \in \mathfrak{P}$ there is a strictly monotonically increasing function $\varphi_{\mathcal{P}} : \mathbb{R}^{\mathcal{P}} \to \mathbb{R}$ such that*

$$H(\mathcal{P}, f_{\mathcal{P}}) = \varphi_{\mathcal{P}}((h_r(f_r))_{r\in\mathcal{P}}),$$

then the reduction principle applies to the minimization of H.

Proof. From monotonicity, $\varphi_{\mathcal{P}}((\min_{f_r} h_r(f_r))_{r\in\mathcal{P}}) \leq \varphi_P((h_r(f_r))_{r\in\mathcal{P}})$ for all functions f_r, $r \in \mathcal{P}$ for all partitions $\mathcal{P}$, we obtain

$$\begin{aligned}\min_{(\mathcal{P},f_{\mathcal{P}})} H(\mathcal{P}, f_{\mathcal{P}}) &= \min_{(\mathcal{P},f_{\mathcal{P}})} \varphi_{\mathcal{P}}((h_r(f_r))_{r\in\mathcal{P}}) \\ &= \min_{(\mathcal{P},f_{\mathcal{P}})} \varphi_{\mathcal{P}}((\min_{f_r} h_r(f_r))_{r\in\mathcal{P}}) \\ &= \min_{\mathcal{P}} \varphi_{\mathcal{P}}((h_r^*)_{r\in\mathcal{P}}).\end{aligned}$$

□

1.2.2 Application to Potts Functionals

In this paragraph we introduce the Potts functional on segmentations and show that the reduction principle applies.

Definition 1.2.5 (Potts functional): *Let a class $\mathfrak{S}$ of segmentations on the image domain S and data $g \in \mathbb{R}^S$ be given. A* **Potts functional** *H_γ with*

parameter $\gamma \geq 0$ *is given by*

$$H_\gamma(\cdot,\cdot) : \mathbb{R}^S \times \mathfrak{S} \longrightarrow \mathbb{R},$$
$$H(g,(\mathcal{P},f_\mathcal{P})) = \gamma \cdot |\mathcal{P}| + \|g - f_\mathcal{P}\|_2^2.$$

The minimization of the Potts functional to data g means finding a γ-weighted balance between the cardinality of the partition $\mathcal{P}$ and the approximation quality $\|f_\mathcal{F} - g\|_2^2$ of $f_\mathcal{F}$. An important observation is the following

Theorem 1.2.6: *Let* $\mathfrak{S}$ *be a segmentation class. The reduction principle applies to the minimization of the Potts functional* $H_\gamma(g,\cdot)$ *over* $\mathfrak{S}$.

Proof. With $\varphi_\mathcal{P}(v) = \sum_{i=1}^{|\mathcal{P}|} \gamma + v_i$ $(v \in \mathbb{R}^\mathcal{P})$ and $h_r(f_r) = \|g|_r - f_r\|_2^2$ $(r \in \mathcal{P})$, the Potts functional has the form $H(g,(\mathcal{P},f_\mathcal{P})) = \varphi_{\mathcal{P},\gamma}((h_r(f_r))_{r\in\mathcal{P}})$. Because $\varphi_\mathcal{P}$ is strictly monotone, the reduction principle applies by Theorem 1.2.4. □

Theorem 1.2.6 tells us: If we fix the partition $\mathcal{P}$, then a minimizer of the Potts functional $H_\gamma(g,(\mathcal{P},\cdot))$ is given by a family of local minimizers $\hat{f}_\mathcal{P} = (\operatorname{argmin}_{f_r\in\mathcal{F}_r} \|g|_r - f_r\|_2^2)_{r\in\mathcal{P}}$. In the sequel we will write $\mathit{\Pi}_{\mathcal{F}_r} g$ for the projection $\hat{f}_r$ of $g|_r$ onto $\mathcal{F}_r$.

1.3 Minimization Algorithms for Potts Functionals

Because the reduction principle applies to Potts functionals H_γ, the complexity of algorithms for a (global) minimization of H_γ is a function of the complexity of the local projections $\mathit{\Pi}_{\mathcal{F}_r}$ $(r \in \mathcal{R})$ and the complexity of the traversing through all admissible partitions. Thus, global minimization algorithms can be developed if, on the one hand, efficient local approximation algorithms are known and if, on the other hand, there is some efficient way to enumerate all possible partitions. For more than one dimension, no such traversing scheme is known for the set of general partitions. Therefore two partition classes that allow an efficient enumeration are introduced in the next chapter.

The algorithmic results in this section will be formulated only for the Potts functional. This permits a shorter notation and a clearer illustration of the basic concepts. Nevertheless the results and proposed algorithms may be easily extended for any functionals fulfilling the reduction principle.

1.3.1 The General Case

Let the image domain S be finite and a segmentation class $\mathfrak{S}(\mathfrak{P}, \mathcal{F})$ on S be given. Assume that the set of admissible fragments is the set of 'used' fragments, $\mathcal{R} = \bigcup_{\mathcal{P}\in\mathfrak{P}} \mathcal{P}$. We present an algorithm for naive minimization of the Potts functional H using the reduction principle.

Algorithm 1.1: Naive Minimization of the Potts functional

input : data z, segmentation class $\mathfrak{S}$, function class $\mathcal{F}$, parameter $\gamma \geq 0$
output: minimizer $(\hat{\mathcal{P}}, \hat{f})$ of H

```
begin
    — preparation —
    foreach r ∈ R do
        σ_r^2 ⟵ ‖Π_{F_r} z − z_r‖_2^2;
    end
    — minimization —
    min ⟵ ∞;
    foreach P ∈ S do
        ssq ⟵ 0;
        foreach r ∈ P do
            ssq ⟵ ssq + σ_r^2;
        end
        h ⟵ ssq + γ · |P|;
        if h < min then
            P̂ ⟵ P; min ⟵ h
        end
    end
    — reconstruction —
    foreach r ∈ P̂ do
        f̂_r ⟵ Π_{F_r} z;
    end
end
```

Notation 1.3.1: *We use the common notation $O(g(n))$ for the order of some function f, i.e. $f(n) = O(g(n))$ implies that there are positive constants $c \in \mathbb{R}_+$ and $k \in \mathbb{N}$ such that $0 \leq |f(n)| \leq cg(n)$ for all $n \geq k$.*

Remark 1.3.2: *In this thesis we will discuss algorithms both in terms of their spatial and temporal complexity. The latter describes the number of necessary basic computational operations to get the result for an input of*

length N. Similarly, the former stands for the number of memory units necessary during the course of the computation. It is common sense that the order of both functionals does not depend on the used computational model. Compare Wagner and Wechsung (1986).

Let C_r be the complexity of the computation of $\|\Pi_{\mathcal{F}_r} z - z_r\|_2^2$, and let C'_r be the complexity of the projection operation $\Pi_{\mathcal{F}_r}$ ($r \in \mathcal{R}$). In this general form the time complexity of the algorithm is given by

$$O(\sum_{r\in\mathcal{R}} C_r + \sum_{r\in\hat{\mathcal{P}}} C'_r + \sum_{\mathcal{P}\in\mathfrak{P}} |\mathcal{P}|).$$

The algorithm has a spatial complexity – a consumption of memory – of $O(|\mathcal{R}|)$.

1.3.2 The One-Dimensional Case

In this paragraph we consider a one-dimensional domain $S = \{1, \ldots, n\}$ and subsets $S_k := \{1, \ldots, k\}$, $1 \le k \le n$. Winkler and Liebscher (2002) propose a dynamic programming algorithm minimizing a Potts functional H with constant regression. We present a slight generalization to larger function spaces.

Let $\mathfrak{S}$ be a set of segmentations of S and $\mathfrak{T}_k = \{(\mathcal{P}, f_\mathcal{P}) \in \mathfrak{S} : \{k, \ldots, n\} \in \mathcal{P}\}$ for each $1 \le k \le n$. The key observation is the following:
$\mathfrak{S} = \bigcup_{k=1}^n \mathfrak{T}_k$ and thus for any functional $H : \mathfrak{S} \to \mathbb{R}$ the following is true

$$\inf_{(\mathcal{P},f_\mathcal{P})\in\mathfrak{S}} H(z, (\mathcal{P}, f_\mathcal{P})) = \min_{1\le k\le n} \inf_{(\mathcal{P},f_\mathcal{P})\in\mathfrak{T}_k} H(z, (\mathcal{P}, f_\mathcal{P})). \tag{1.2}$$

If the reduction principle holds for H, then the minimization can additionally be split into global minimization over partitions and local projections onto function spaces. To avoid a bloat in notation, we illustrate this with the Potts functional in the following Lemma, although a more general result could be formulated. We use the short notation $z_{[k,l]} := z|_{[k,l]\cap\mathbb{N}}$ for restricted data and $f_{[k,l]} := f_{[k,l]\cap\mathbb{N}}$ for functions in the function spaces $\mathcal{F}_{[k,l]} := \mathcal{F}_{[k,l]\cap\mathbb{N}}$.

Lemma 1.3.3: *Let $\mathcal{F}$ be a class of admissible functions on the image domain S, $\mathcal{R}$ the set of intervals in S, and $\mathfrak{S}_k$ the corresponding segmentations of S_k ($1 \le k \le n$). Assume that H is the Potts functional and set*

$$B_{z,\gamma}(k) := \inf_{(\mathcal{P},f_\mathcal{P})\in\mathfrak{S}_k} H_\gamma(z|_{S_k}, (\mathcal{P}, f_\mathcal{P})) \, (1 \le k \le n),$$

$B_{z,\gamma}(0) := 0$ and $\hat{f}_{[k,n]} := \Pi_{\mathcal{F}_{[k,n]}} z$. Then the following holds:

$$B_{z,\gamma}(j) = \min_{1 \le k \le j} \left(B_{z,\gamma}(k-1) + \gamma + \|\hat{f}_{[k,j]} - z_{[k,n]}\|_2^2 \right)$$

for all $1 \le j \le n$.

Proof. Let a partition $\mathcal{P} = \mathcal{P}_{k-1} \cup \{\{k, \ldots, n\}\}$ be given, then

$$\begin{aligned} H_\gamma(z, (\mathcal{P}, f_\mathcal{P})) &= \gamma \cdot |\mathcal{P}| + \|z - f_\mathcal{P}\|_2^2 \\ &= \gamma \cdot (|\mathcal{P}_{k-1}| + 1) + \|z|_{S_{k-1}} - f_{\mathcal{P}_{k-1}}\|_2^2 + \|z_{[k,n]} - f_{[k,n]}\|_2^2 \\ &= H_\gamma(z|_{S_{k-1}}, (\mathcal{P}_{k-1}, f_{\mathcal{P}_{k-1}})) + \gamma + \|z_{[k,n]} - f_{[k,n]}\|_2^2. \end{aligned}$$

Because the reduction principle holds for a Potts functional by Theorem 1.2.6, applying equation 1.2 completes the proof. □

The previous Lemma allows one to write down the algorithm immediately. The data structure used for storing one-dimensional partitions in Algorithm 1.2 is an array p with length $n+1$ starting at 0. At position $1 \le l \le n$, the array p contains the best previous position: $p_l = \operatorname{argmin}_{1 \le k \le l}(B_{z,\gamma}(k-1) + \gamma + \|\hat{f}_{[k,n]} - z_{[k,n]}\|_2^2)$, $p_0 := -1$ is used as a sentinel for stopping the projection loop.

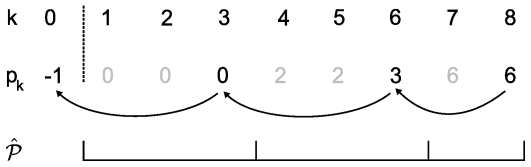


Fig. 1.6: Data structure for one-dimensional partitions used in Algorithm 1.2

The preparation loop used in Algorithm 1.1 for storing the L^2 distance of the projections to the signal is not necessary in Algorithm 1.2. The computation can be done in the minimization loop because therein each interval is processed only once.

Algorithm 1.2: Minimization of the one-dimensional Potts functional H_γ for fixed $\gamma \geq 0$

input : data z, function class $\mathcal{F}$, parameter $\gamma \geq 0$
output: minimizer $(p, \hat{f})$ of H

```
begin
    — minimization—
    ĥ_0 := 0; p_0 := −1;
    for k ← 1 to n do
        ĥ_k ← ∞;
        for l ← 1 to k do
            σ² ← ‖Π_{F_[l,k]} z − z_[l,k]‖₂²;
            h ← ĥ_{l−1} + σ²;
            if h < ĥ_k then
                ĥ_k ← h;
                p_k ← l − 1;
            end
        end
    end
    — reconstruction —
    k ← n; l ← p_k;
    while l ≥ 0 do
        f̂_[l+1,k] ← Π_{F_[l+1,k]} z;
        k ← l; l ← p_k;
    end
end
```

For Algorithm 1.2 the following complexity result holds:

Lemma 1.3.4: *Assume for an interval $r = [l, k] \cap \mathbb{N}$ $(1 \leq k \leq l \leq n)$ in S that the complexity of the computation of $\|\Pi_{\mathcal{F}_r} z - z_r\|_2^2$ is given by $C_1(|r|)$, and let $C_2(|r|)$ be the complexity for the projection operation $\hat{f}_r \leftarrow \Pi_{\mathcal{F}_r} z$. Then the time complexity of Algorithm 1.2 is given by*

$$O\left(\sum_{m=1}^{n} m \cdot C_1(n-m+1) + \sum_{r \in \hat{\mathcal{P}}} C_2(|r|)\right).$$

Proof. It only has to be shown that $\sum_{k=1}^{n}\sum_{l=1}^{k} C_1(k-l+1) = \sum_{m=1}^{n} m \cdot C_1(n-m+1)$:

$$\begin{aligned}\sum_{k=1}^{n}\sum_{l=1}^{k} C_1(k-l+1) &= \sum_{k=1}^{n}\sum_{l'=1}^{k} C_1(l') = \sum_{m=1}^{n} |\{k : k \geq m\}| \cdot C_1(m) \\ &= \sum_{m=1}^{n}(n-m+1)C_1(m) = \sum_{m=1}^{n} mC_1(n-m+1),\end{aligned}$$

which completes the proof by identification of these sums with the nested loop in the algorithm. □

The following is a consequence of the previous lemma.

Corollary 1.3.5: *If, with the assumptions of the previous Lemma, $C_1(\cdot)$ is constant and $C_2(\cdot)$ is linear, then Algorithm 1.2 has a time complexity of $O(n^2)$. The spatial complexity of Algorithm 1.2 is of $O(n)$.*

Proof. Let $C_1(m) = c_1$ for all m, and let $C_2(m) = m \cdot c_2$ for all $1 \leq m \leq n$, then $\sum_{m=1}^{n} mc_1 + \sum_{r \in \hat{\mathcal{P}}} |r| c_2 = c_1 \frac{n(n+1)}{2} + c_2 n$. For storing the minimizing partition in Algorithm 1.2, an array $\hat{h}$ of length n and an array p of length $n+1$ is used. □

There is a very similar algorithm performing the minimization of the Potts functional in one dimension for all $\gamma \in \mathbb{R}$ simultaneously. Because its analysis does not reveal anything concerning the geometric structure of the minimization problem, we do not comment on it here. See Winkler and Liebscher (2002) and Kempe (2003).

2. Partitions allowing Efficient Recursion

In the previous chapter, the following two features were needed for an efficient minimization of the Potts functional. First, fast algorithms for the local regressions are fundamental for the efficiency of the minimization operation ('inner loop'). This vital point will be addressed in Chapter 3. Second, schemes for a rapid traversing through all partitions in a partition class are required ('outer loop'). In the one-dimensional case there is such a procedure for the most general partition class, which processes each interval of S exactly once. Because the number of (discrete) intervals of $\{1, \ldots, N\}$ is $N(N+1)/2$, algorithms of this kind are reasonable. For more than one dimension, however, this does not make any sense. There is no procedure known to us that traverses the full class of partitions of the two-dimensional image domain $S = \{1, \ldots, N_1\} \times \{1, \ldots, N_2\}$ where each partition element is processed only once. Moreover, the set of fragments of S is huge. The class of partitions has to be restricted in such a way that it allows traversing it with acceptable complexity while – together with the local projections – still producing reasonable approximations.

In Section 2.1, a partition class is presented that allows the use of Algorithm 1.2 proposed in Subsection 1.3.2 for more than one dimension. This partition class has a structure that allows Algorithm 1.2 to be applied recursively.

In Section 2.2, a partition class with a quad-tree structure is introduced, which permits very fast traversing.

The partitions presented in this chapter will be defined for two dimensions. In the continuous case for the two-dimensional image domain, we take the half open square $[0, 1)^2$. It is only notational simplicity not to take $[0, 1]^2$ as image domain in that partitions consisting of rectangles can be introduced much easier when all rectangles have the same shape.

2.1 Hierarchic Partitions

In this section, we propose a class of partitions that are formed by recursion over the dimensions of the image domain S. The minimization of a functional fulfilling the reduction principle can be performed by recursively applying a dynamic programming algorithm, as proposed in Subsection 1.3.2. Although the proposed partition class could be defined in any dimension, for notational simplicity we give a definition for an index set with dimension two.

2.1.1 Definition of the Partition Class

In the following, the image domains are either given by

1. $S_1 = \{1, \dots, N_1\}$, $S_2 = \{1, \dots, N_2\}$ and $S = S_1 \times S_2$, or by
2. $S_1 = S_2 = [0, 1)$ and $S = [0, 1)^2$.

Lemma 2.1.1: *Let $(\mathcal{I}_i)_{1 \le i \le n}$ be a partition of S_1 with $n \in \mathbb{N}$ elements, and let for each $1 \le i \le n$ the family $(\mathcal{I}_{ij})_{1 \le j \le m_i}$ ($m_i \in \mathbb{N}$) be a partition of S_2. Assume*

$$r_{ij} := \mathcal{I}_i \times \mathcal{I}_{ij}, \quad 1 \le j \le m_i,\, 1 \le i \le n.$$

Then the set

$$\mathcal{P} = \{r_{ij} : 1 \le j \le m_i, 1 \le i \le n\} \tag{2.1}$$

is a partition of $S_1 \times S_2$. Moreover, the set

$$\mathcal{P}_i := \{r_{ij} : 1 \le j \le m_i\} \tag{2.2}$$

is a partition of $\mathcal{I}_i \times S_2$ for each $1 \le i \le n$.

Proof. Conditions (A1) - (A3) from Definition 1.1.1 are easily verified: Firstly,

$$\bigcup_{(i,j)} \mathcal{I}_i \times \mathcal{I}_{ij} = \bigcup_{i=1}^{n} \bigcup_{j=1}^{m_i} \mathcal{I}_i \times \mathcal{I}_{ij} = \bigcup_{i=1}^{n} \mathcal{I}_i \times S_2 = S_1 \times S_2.$$

Secondly,

$$\mathcal{I}_i \times \mathcal{I}_{ij} \cap \mathcal{I}_{i'} \times \mathcal{I}_{i'j'} = \emptyset \text{ if } i \neq i' \text{ or } j \neq j'.$$

And thirdly, rectangles (in $\mathbb{Z}^2$ and $\mathbb{R}^2$) are connected. □

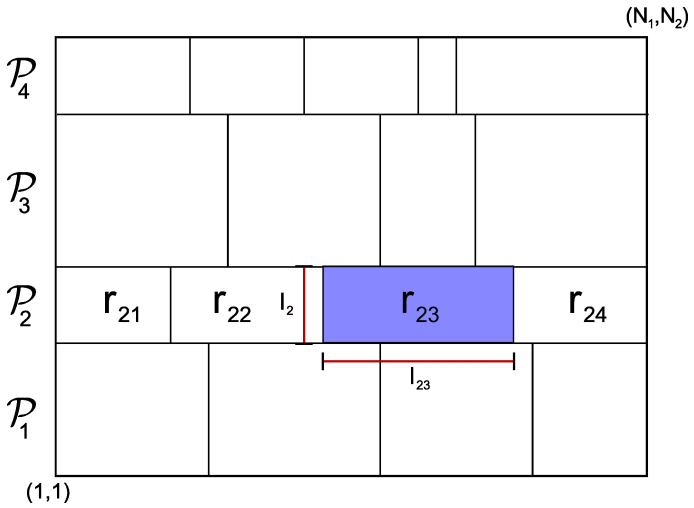


Fig. 2.1: Hierarchic Partition

Lemma 2.1.1 justifies the following definition.

Definition 2.1.2 (Hierarchic Partition): *A partition of the form (2.1) is called a* **hierarchic partition** *of S.*

A hierarchic partition of an image domain is generated by the specification of a partition $\mathcal{P}^{(1)}$ of the – say – x-axis, and then partitioning the other direction for each of the fragments of $\mathcal{P}^{(1)}$.

2.1.2 Continuous versus Discrete Domain

In the previous paragraph, hierarchic partitions have been defined simultaneously for both continuous and discrete image domains. To establish a link between the continuous and discrete cases, we introduce a canonical discretization of subsets of $[0,1)^2$ to subsets of $S = \{1,\ldots,N_1\} \times \{1,\ldots,N_2\}$. With a view towards discretization of signals (real valued functions) over $[0,1)^2$, a natural choice to discretize $[0,1)^2$ is the division of $[0,1)^2$ into $N_1 \cdot N_2$ tiles of size $\frac{1}{N_1} \times \frac{1}{N_2}$. Taking the midpoints of these tiles as reference points results in the following discretization of subsets of $[0,1)^2$ to subsets of S.

Notation 2.1.3: *Consider a discrete rectangle $R = [s_1, s_1+n_1) \times [s_2, s_2+n_2) \cap \mathbb{Z}^2$, $s_1, s_2 \in \mathbb{Z}$, $n_1, n_2 \in \mathbb{N}$, and a set $A \subset [0,1)^2$. By $\boldsymbol{\delta}^R(A)$ we denote the discretization of A in R defined by*

$$\boldsymbol{\delta}^R(A) = \{(s_1 - \frac{1}{2} + n_1 \cdot a_1, s_2 - \frac{1}{2} + n_2 \cdot a_2) : (a_1, a_2) \in A\} \cap \mathbb{Z}^2.$$

Let $\mathcal{P}$ be a partition of $[0,1)^2$. For notational convenience we use the same symbol $\boldsymbol{\delta}^R$ for the discretization of a partition:

$$\boldsymbol{\delta}^R(\mathcal{P}) := \{\boldsymbol{\delta}^R(A) : A \in \mathcal{P}\}.$$

Note that $\boldsymbol{\delta}^R(\mathcal{P})$ is not necessarily a partition (see Remark 1.1.2).

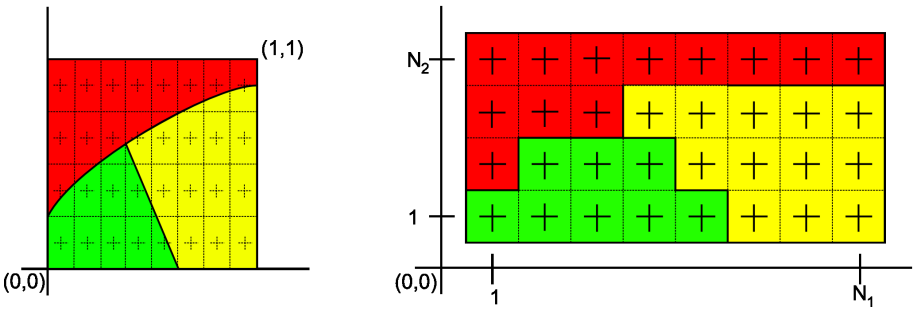


Fig. 2.2: Partition of $[0,1)^2$ (left) and canonical discretization $\boldsymbol{\delta}^R$ with $R = \{1,\dots,8\} \times \{1,\dots,4\}$ (right).

Assume that $s,t \in [0,1)$, $s \le t$ and $N \in \mathbb{N}$. The discrete set

$$\{N \cdot x + \frac{1}{2} : x \in [s,t)\} \cap \mathbb{Z} = [N \cdot s + \frac{1}{2}, N \cdot t + \frac{1}{2}) \cap \mathbb{Z}$$

is either connected or empty. This observation (which is also true for the intervals (s,t), $(s,t]$ and $[s,t]$) leads to the following:

Lemma 2.1.4: *If a partition $\mathcal{P}$ of $[0,1)^2$ consists of rectangles, the discretization $\boldsymbol{\delta}^S(\mathcal{P})$ of $\mathcal{P}$ is a partition of S.*

Proof. Firstly, $S = \bigcup_{r\in\mathcal{P}} \boldsymbol{\delta}^S(r)$. Secondly, if subsets r and q of $[0,1)^2$ are disjoint, then $\boldsymbol{\delta}^S(r)$ and $\boldsymbol{\delta}^S(q)$ are also disjoint. Thirdly, let, without loss of generality, the rectangle r be of the form $r = [s_1,t_1) \times [s_2,t_2)$, then the set

$$\begin{aligned}\boldsymbol{\delta}^S(r) &= \{(N_1 s_1 + \frac{1}{2}, N_2 s_2 + \frac{1}{2}) : (s_1,s_2) \in r\} \cap \mathbb{Z}^2 \\ &= [N_1 s_1 + \frac{1}{2}, N_1 t_1 + \frac{1}{2}) \times [N_2 s_2 + \frac{1}{2}, N_2 t_2 + \frac{1}{2}) \cap \mathbb{Z}^2\end{aligned}$$

is either connected or empty. □

The canonical discretization even preserves some more structure. With $\boldsymbol{\delta}^S$, hierarchic partitions are transformed into hierarchic partitions:

Lemma 2.1.5: *The discretization $\boldsymbol{\delta}^S(\mathcal{P})$ of a hierarchic partition $\mathcal{P}$ of $[0,1)^2$ is a hierarchic partition of $S = \{1,\dots,N_1\} \times \{1,\dots,N_2\}$.*

Proof. Let with the assumptions of Lemma 2.1.1 (for a continuous image domain) $r_{ij} = \mathcal{I}_i \times \mathcal{I}_{ij} \subset [0,1)^2$, then

$$\begin{aligned}\boldsymbol{\delta}^S(r_{ij}) &= \boldsymbol{\delta}^S(\mathcal{I}_i \times \mathcal{I}_{ij}) \\ &= \{N_1 s_1 + \frac{1}{2} : s_1 \in \mathcal{I}_i\} \cap \mathbb{Z} \times \{N_2 s_2 + \frac{1}{2} : s_2 \in \mathcal{I}_{ij}\} \cap \mathbb{Z} \\ &=: \mathcal{I}'_i \times \mathcal{I}'_{ij}.\end{aligned}$$

The family $\{\mathcal{I}'_i : 1 \le i \le n\}$ is a partition of $\{1,\dots,N_1\}$, and $\{\mathcal{I}'_{ij} : 1 \le j \le m_i\}$ is a partition of $\{1,\dots,N_2\}$ for each $1 \le i \le n$. Thus, the set $\boldsymbol{\delta}^S(\mathcal{P}) = \{\boldsymbol{\delta}^S(r_{ij}), 1 \le j \le m_i, 1 \le i \le n\}$ is a hierarchic partition. □

Remark 2.1.6: *Hierarchic partitions can be defined for an arbitrary number of dimensions in the following way. Assume the (discrete) index set S is a cuboid with $d \in \mathbb{N}$ dimensions and side lengths $N_i \in \mathbb{N}$, $1 \le i \le n$. Let, for all $1 \le k \le d$, the symbol $\mathfrak{I}_k$ denote the set of all partitions of $\{1,\dots,N_k\}$. The class of recursive partitions $\mathfrak{P}_d$ of S is recursively defined by*

$$\mathfrak{P}_1 = \mathfrak{I}_1$$
$$\mathfrak{P}_k = \{\{r \times q : q \in \mathcal{P}_r, r \in \mathcal{Q}\} : \mathcal{P}_r \in \mathfrak{P}_{k-1} \; \forall\, r \in \mathcal{Q}, \mathcal{Q} \in \mathfrak{I}_k\}, \quad 1 < k \le d.$$

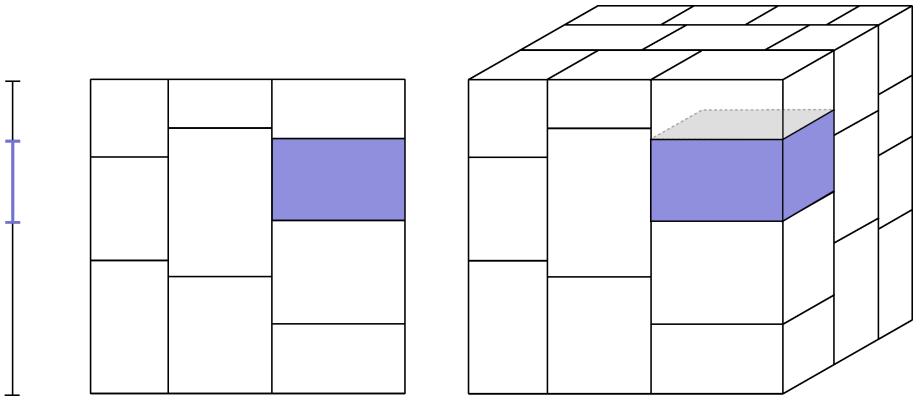


Fig. 2.3: *Elements of $\mathfrak{P}_1$, $\mathfrak{P}_2$ and $\mathfrak{P}_3$*

2.1.3 Recursion

We make use of the recursive structure of the hierarchic partitions and present an algorithm for minimizing the two-dimensional Potts model. First we de-

velop the recursion formulas.

Consider $S_1 = \{1, \dots, N_1\}$, $S_2 = \{1, \dots, N_2\}$, $S = S_1 \times S_2$, and let $\mathfrak{P}$ be the class of partitions of S, $\mathcal{F}$ the class of admissible functions and $\mathfrak{S}$ the resulting class of segmentations. Assume data $z \in \mathbb{R}^S$ and $h_r^* = \inf_{f_r \in \mathcal{F}_r} \|z|_r - f_r\|_2^2$. We define a functional

$$K_{\mathcal{P}} := \gamma|\mathcal{P}| + \sum_{p \in \mathcal{P}} h_r^*, \ \mathcal{P} \in \mathfrak{P}.$$

Then for the Potts functional H_γ with fixed data z and $\gamma \in \mathbb{R}$ the following holds:

$$\inf_{(\mathcal{P}, f_{\mathcal{P}}) \in \mathfrak{S}} H_\gamma(z, (\mathcal{P}, f_{\mathcal{P}})) = \min_{\mathcal{P} \in \mathfrak{P}} K_{\mathcal{P}}.$$

Let, with the assumptions of Lemma 2.1.1, $\mathcal{P}$ be a partition of the form (2.1). Observe that

$$K_{\mathcal{P}} = \gamma|\bigcup_{i=1}^{n}\bigcup_{j=1}^{m_i} r_{ij}| + \sum_{i=1}^{n}\sum_{j=1}^{m_i} h_{r_{ij}}^* = \sum_{i=1}^{n} K_{\mathcal{I}_i \times \mathcal{P}_i}.$$

Assume that $\mathfrak{P}(n)$ is the set of partitions of $\{1, \dots, n\}$, $n \in \mathbb{N}$. Then for the minimization of K we have

$$\min_{\mathcal{P} \in \mathfrak{P}} K_{\mathcal{P}} = \min_{I \in \mathfrak{P}(N_1)} \sum_{i=1}^{n_I} \min_{\mathcal{P}_i \in \mathfrak{P}(N_2)} K_{I_i \times \mathcal{P}_i}.$$

For each interval I of S_2 consider $K_I^* = \min_{\mathcal{P} \in \mathfrak{P}(N_2)} K_{I \times \mathcal{P}}$. We therefore have a recursion formula similar to that of Lemma 2.1.1:

$$\min_{I \in \mathfrak{P}(n)} \sum_{i=1}^{n_I} K_{I_i}^* = \min_{1 \le k \le n} \min_{I \in \mathfrak{P}(k)} \sum_{i=1}^{n_I} K_{I_i}^* + K_{\{k,\dots,n\}}^*, \ 1 \le n \le N_1. \tag{2.3}$$

Moreover, for the computation of K_I^* we have

$$\min_{\mathcal{P} \in \mathfrak{P}(n)} K_{I \times \mathcal{P}} = \min_{1 \le k \le n} \min_{\mathcal{P} \in \mathfrak{P}(k)} K_{I \times \mathcal{P}} + h_{I \times \{k,\dots,n\}}^*, \ 1 \le n \le N_2. \tag{2.4}$$

Formulas (2.3) and (2.4) give rise to the following algorithm.

The algorithm consists of nested loops for dimension 1 and 2. To permit a concise display of the algorithm, the inner loop corresponding to (2.3) is presented as a procedure that is called from within the outer loop according to equation (2.4) in the algorithm. To avoid an overhead of technical details, we wrap the global parameters used in both the inner and outer loop in an abstract reference object REF of generic type OBJECT. In a computer language this is typically implemented with composite types such as

records, structs, sets, classes or objects, together with a pointer mechanism that locates a potentially repository for large data. The abstract entry 'function class $\mathcal{F}$' could be realized by fixed procedures, type bound procedures (methods), procedure variables, etc. We access the content of the reference object by the common 'REF.entry' notation.

Consider an interval $I = [f, t] \cap \mathbb{Z}$ of S_1. Let $\mathfrak{P}_2$ be the set of partitions of S_2. Now we present a procedure that does the minimization of the Potts functional H for all partitions of the form $\{I \times r : r \in \mathcal{P}_2\}$, $\mathcal{P}_2 \in \mathfrak{P}_2$.

Procedure `StripeMin`($f, t \in \mathbb{N}$; REF: *OBJECT*)

input : function class $\mathcal{F}$, $z \in \mathbb{R}^S$ and $\gamma \in \mathbb{R}$ stored in reference object REF
output: minimum of H and minimizing partition $\hat{\mathcal{P}}$ of H stored in REF

```
begin
    z ← REF.z; γ ← REF.γ; F ← REF.F;
    ĥ_0 := 0; P̂_0 := −1; n := N_2;
    for k ← 1 to n do
        ĥ_k ← ∞;
        for l ← 1 to k do
            σ² ← ‖Π_{F_[f,t]×[l,k]} z − z_{[f,t]×[l,k]}‖₂²;
            h ← ĥ_{l−1} + σ²;
            if h < ĥ_k then
                ĥ_k ← h;
                P̂_k ← l − 1;
            end
        end
    end
    REF.P̂ ← P̂; REF.ĥ ← ĥ_n; REF.from ← f; REF.to ← t;
end
```

Assume that the minimization procedure above has been performed, and REF contains the best partition for some interval $I = [\text{REF}.from, \text{REF}.to]$. The following procedure can then be used to do the resulting local projections.

Procedure `StripeReconstruction`(REF: *OBJECT*)

input : function class $\mathcal{F}$, $from, to \in \mathbb{N}$, $z \in \mathbb{R}^S$ and $\gamma \in \mathbb{R}$ stored in reference object REF
output: Projection $\hat{f}$ according to $\text{REF}.\hat{\mathcal{P}}$ stored in REF

```
begin
    z ← REF.z; f ← REF.from; t ← REF.to; n := N_2; p ← REF.P̂;
    k ← n; l ← p_k;
    while l ≥ 0 do
        f̂_{[l+1,k]} ← Π_{F_[f,t]×[l+1,k]} z;
        k ← l; l ← p_k;
    end
    REF.f̂ ← f̂;
end
```

Algorithm 2.3: Minimization of the two-dimensional Potts functional H_γ over hierarchic partitions for fixed $\gamma \geq 0$

```
input  : data z, function class F, parameter γ ≥ 0
output: minimizer f̂ of H
begin
    — preparation —
    Ref.γ ← γ; Ref.z ← z; Ref.F ← F;
    — minimization—
    ĥ_0 := 0; P_0 := −1; n := N_1;
    for k ← 1 to n do
        ĥ_k ← ∞;
        for l ← 1 to k do
            StripeMin (k,l,Ref);
            σ² ← Ref.h;
            h ← ĥ_{l−1} + σ²;
            if h < ĥ_k then
                ĥ_k ← h;
                p_k ← l − 1;
            end
        end
    end
    — reconstruction —
    k ← n; l ← p_k;
    while l ≥ 0 do
        StripeMin (l+1,k,Ref);
        StripeReconstruction(Ref);
        f̂_{[l+1,k]} ← Ref.f̂;
        k ← l; l ← p_k;
    end
end
```

The repeated call of `StripeMin` in the reconstruction loop is due to the fact that the partitions of the local reconstructions are not stored in the minimization loop. This way the algorithm has a memory consumption of $O(N_1 + N_2)$. A storage of the local minimizing partitions would mean a memory usage of $O(N_1 \cdot N_2)$, and a bit more administrative overhead. In the following, by $\hat{\mathcal{P}}^{(1)}$ we denote the result of the outer minimization loop.

For Algorithm 2.3, the following complexity result holds:

Lemma 2.1.7: *Assume for a rectangle r in S with side lengths w and h that the complexity of the computation of $\|\Pi_{\mathcal{F}_r} z - z_r\|_2^2$ is given by $C_1(w,h)$, and let $C_2(|r|)$ be the complexity for the projection operation $\hat{f}_r \leftarrow \Pi_{\mathcal{F}_r} z$. Then*

the complexity of the minimization part of Algorithm 1.2 is given by

$$O\left(\sum_{l=1}^{N_1}\sum_{n=1}^{N_2} l\cdot n\cdot C_1((N_1-l+1),(N_2-n+1))\right),$$

and the reconstruction loop has a complexity of

$$O\left(\sum_{p\in\hat{\mathcal{P}}^{(1)}}\left(\sum_{n=1}^{N_2} n\cdot C_1(|p|,(N_2-n+1))+\sum_{p'\in\hat{\mathcal{P}}_r} C_2(|p|\cdot|p'|)\right)\right).$$

Proof. For the statement concerning the minimization part, we need the following equality derived in the same way as in the proof of Lemma 1.3.4:

$$\begin{aligned}\sum_{k=1}^{N_1}\sum_{l=1}^{k}\sum_{m=1}^{N_2}\sum_{n=1}^{m} C_1((k-l+1),(m-n+1)) &= \sum_{k=1}^{N_1}\sum_{l=1}^{k}\sum_{m=1}^{N_2}\sum_{n=1}^{m} C_1(l,n)\\ &= \sum_{l=1}^{N_1}\sum_{n=1}^{N_2} l\cdot n\cdot C_1((N_1-l+1),(N_2-n+1)).\end{aligned}$$

The following equality is required for the statement regarding the reconstruction loop:

$$\begin{aligned}&\sum_{p\in\hat{\mathcal{P}}^{(1)}}\left(\sum_{k=1}^{N_2}\sum_{n=1}^{k} C_1(|p|,(N_2-n+1))+\sum_{p'\in\hat{\mathcal{P}}_r} C_2(|p|\cdot|p'|)\right)\\ &\qquad=\sum_{r\in\hat{\mathcal{P}}^{(1)}}\left(\sum_{n=1}^{N_2} n\cdot C_1(|p|,(N_2-n+1))+\sum_{p'\in\hat{\mathcal{P}}_r} C_2(|p|\cdot|p'|)\right).\end{aligned}$$

The different terms of the above sum correspond to the decomposition of the algorithm into its nested loops, which completes the proof. □

Lemma 2.1.7 indicates that the complexity of the minimization process is crucially dependent on the efficiency of the local projections. The complexity of Algorithm 2.3 for projections with maximal efficiency is given in the following

Corollary 2.1.8: *If, with the assumptions of the previous Lemma, there is some $c_1 > 0$, which may depend on other variables than the size, such that $C_1(w,h) = c_1(w+h)$ and $C_2(\cdot)$ is linear, then Algorithm 2.3 has complexity $O(c_1\cdot|S|^2(N_1+N_2))$.*

Proof. Let $C_2(m) = m \cdot c_2$ for all $m \in \mathbb{N}$. Then, by

$$\sum_{l=1}^{N_1}\sum_{n=1}^{N_2} l \cdot n \cdot c_1(N_1 - l + N_2 - n + 2)$$
$$\leq c_1(2 + N_1 + N_2)(\frac{N_1(N_1+1)}{2} + \frac{N_2(N_2+1)}{2})$$

the minimization loop has complexity $O(c_1 N_1^2 N_2^2 (N_1 + N_2))$, and by

$$\sum_{p\in\hat{\mathcal{P}}^{(1)}} \left(\sum_{n=1}^{N_2} c_1 n(|p| + N_2 - n + 1) + \sum_{p'\in\hat{\mathcal{P}}_r} |p| \cdot |p'| \cdot c_2 \right)$$
$$\leq (N_1 + N_2 + 1)\frac{N_2(N_2+1)}{2} c_1 + N_1 \cdot N_2 \cdot c_2$$

the reconstruction has complexity $O(c_1(N_1 + N_2)N_2^2)$. □

2.2 Dyadic Partitions

In this section we define another class of partitions with a different kind of recursive structure. Dyadic partitions will be introduced by a restriction on the set of admissible fragments $\mathcal{R}$. At a first glance, the recursive structure of dyadic partitions will not be readily identifiable; therefore, it will need to be devised subsequently. At last an efficient algorithm for traversing each element in the partition class is given.

2.2.1 Continuous Image Domain

A unified approach to both the discrete and continuous domains cannot be acquired as easily as in the last section. The reason for this is that, unlike the hierarchic case, we cannot directly utilize an analogy of the form of discrete and continuous partitions in one dimension. Therefore, until further notice, let the image domain be continuous, $S = [0, 1)^2$. We start with the definition of dyadic partitions straightaway.

Definition 2.2.1 (Dyadic Square, Dyadic Partition, Depth): *Consider for each $n \in \mathbb{N}$ the set*

$$\mathcal{D}_{-n} = \{[(i-1)\cdot 2^{-n}, i\cdot 2^{-n}) \times [(j-1)\cdot 2^{-n}, j\cdot 2^{-n}) : 1 \leq i, j \leq 2^n\}.$$

An element of the family $\mathcal{D} = \bigcup_{n\in\mathbb{N}} \mathcal{D}_{-n}$ *will be called a* **dyadic square**.

A partition $\mathcal{P} \subset \mathcal{D}$ *of* S *(consisting of dyadic squares) is called a* **dyadic partition**.

We will say that a dyadic partition $\mathcal{P}$ **has depth** $n \in \mathbb{N}$ *if* n *is the smallest integer such that* $\mathcal{P} \subset \bigcup_{k=0}^{n} \mathcal{D}_{-n}$.

Before we can reveal the recursive structure of dyadic partitions we need some preparation. In the following definition for finite sets A and B, the notation $A \boxtimes B = \{a \times b : a \in A, b \in B\}$ is used.

Definition 2.2.2: *Let* $r = [s_1, t_1) \times [s_2, t_2)$ *be a rectangle in* S*, and let* (m_1, m_2) *be the midpoint* $(m_1 = \frac{t_1+s_1}{2},\ m_2 = \frac{t_2+s_2}{2})$*. A* **quad split** *of* r *is the following set of four rectangles*

$$\texttt{qsplit}(r) = \{[s_1, m_1), [m_1, t_1)\} \boxtimes \{[s_2, m_2), [m_2, t_2)\}.$$

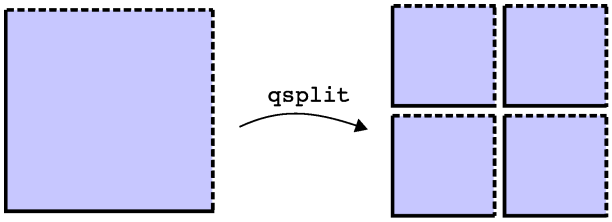


Fig. 2.4: quad split

Remark 2.2.3: *Let* $\mathcal{P}$ *be a partition of* S*. For* $q \in \mathcal{P}$*, let the set* $\mathcal{Q}$ *be a partition of* q*. Then the family* $\mathcal{P}' = \mathcal{P} \cup \mathcal{Q} \setminus q$ *generated by replacing* q *in* $\mathcal{P}$ *by* $\mathcal{Q}$ *is also a partition of* S*. Moreover, the quad split of a rectangle* r *is a partition of* r*. Therefore, replacing some rectangle* q *in a partition by the quad split of* q *results in a partition.*

Now we define a recursive structure.

Definition 2.2.4: *Fix a number* $n \in \mathbb{N}$*. The pair* (V_n, E_n) *defined recursively by* $V_0 = \{S\}$*,* $E_0 = \emptyset$ *and*

$$V_k = V_{k-1} \cup \texttt{qsplit}(r_{k-1}),$$
$$E_k = E_{k-1} \cup \{(r_{k-1}, q) : q \in \texttt{qsplit}(r_{k-1})\},$$

for $1 \le k \le n$ *with* $r_{k-1} \in V_{k-1}$*, is a tree. We will call such a tree a* **quad-tree** *induced by* $(r_k)_{0\le k<n}$.

The link to partitions becomes clear with the following statement:

Lemma 2.2.5: *The set of terminal nodes of a quad-tree* (V_n, E_n),

$$L(V_n) = \{v \in V_n : \texttt{qsplit}(v) \not\subset V\}, \tag{2.5}$$

is a partition of S.

Proof. We prove the statement by induction: $V_0 = L(V_0)$ is a partition of S. Assume $L(V_k)$ is a partition of S, then $L(V_{k+1}) = L(V_k) \setminus r_k \cup \texttt{qsplit}(r_k)$ is also a partition by Remark 2.2.3. □

We will denote a partition consisting of the terminal nodes of a quad-tree a **quad-tree partition**.

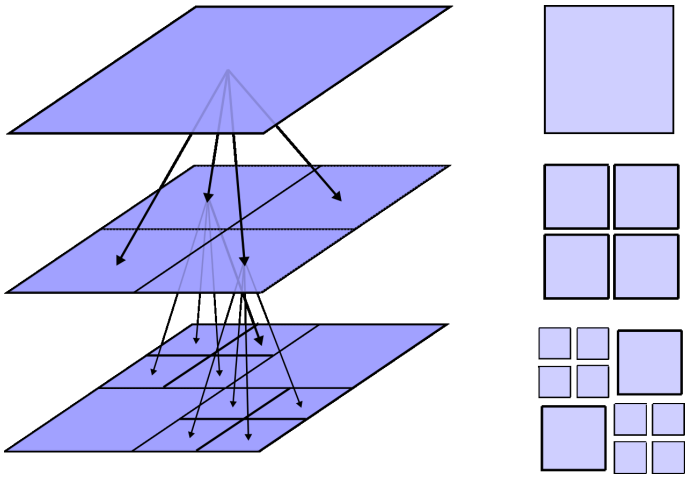


Fig. 2.5: Illustration of a quad-tree

With the following preparation it is easy to prove that quad-tree partitions consist of dyadic squares.

Lemma 2.2.6: *Let* r *be a dyadic square. Then* $\texttt{qsplit}(r)$ *consists of dyadic squares.*

Proof. Assume $r = [(i-1)2^{-l}, i2^{-l}) \times [(j-1)2^{-l}, j2^{-l})$ $(1 \leq i, j \leq 2^l$, $l \in \mathbb{N})$, then the midpoint of r is given by $((2i-1)2^{-l-1}, (2j-1)2^{-l-1})$.

With $r = [(2i-2)2^{-l-1}, 2i \cdot 2^{-l-1}) \times [(2j-2)2^{-l-1}, 2j \cdot 2^{-l-1})$, the rest is clear by definition. □

Lemma 2.2.7: *A quad-tree partition $\mathcal{P}$ consists of dyadic squares.*

Proof. We prove the statement by induction. $V_0 = [0,1)^2$ is a dyadic square. Let V_k consist of dyadic squares. Then $V_{k+1} = V_k \cup \texttt{qsplit}(r)$ for some dyadic square $r \in V_k$. By Lemma 2.2.6, $\texttt{qsplit}(r)$ consists of dyadic squares, and because $L(V_{k+1}) \subset V_{k+1}$ the proof is complete. □

It will become clear that not only quad-tree partitions are dyadic square partitions, but that also the converse is true: dyadic partitions are quad-tree partitions. For the proof we need the following observation:

Lemma 2.2.8: *For any dyadic square r there is a number $n \in \mathbb{N}$ and a sequence of dyadic squares $(r_i)_{1 \le i \le n}$ such that $r_0 = [0,1)^2$, $r_n = r$ and $r_k \in \texttt{qsplit}(r_{k-1})$ for all $1 \le k \le n$.*

Proof. We start with proving the analogous result for one dimension by induction. Let a dyadic interval $I = [(i-1)2^{-k}, i2^{-k})$, $1 \le i \le k$, $k \in \mathbb{N}$ be given. Consider the case $i = 2j$ for some $j \in \mathbb{N}$. Then $1 \le j \le 2^{k-1}$ and $I = [(j-1)2^{-(k-1)} + \frac{2^{-(k-1)}}{2}, j2^{-(k-1)})$, the right half of the interval being $[(j-1)2^{-(k-1)}, j2^{-(k-1)})$. Now consider $i = 2j-1$ for some $j \in \mathbb{N}$. Again, $1 \le j \le 2^{k-1}$ and $I = [(j-1)2^{-(k-1)}, j2^{-(k-1)} - \frac{2^{-(k-1)}}{2})$, and the left half of the interval is $[(j-1)2^{-(k-1)}, j2^{-(k-1)})$. Carried forward to the two-dimensional case, this means that for each dyadic rectangle r there is some dyadic rectangle r' with doubled side length such that $r \in \texttt{qsplit}(r)$. If the side length of r is 2^{-k}, then $n = k$. □

Lemma 2.2.9: *Let $\mathcal{P}$ be a dyadic partition of $[0,1)^2$, then $\mathcal{P}$ is a quad-tree partition of $[0,1)^2$.*

Proof. By the previous lemma, for each $p \in \mathcal{P}$ there exists a sequence $(r_k^p)_{0 \le k \le n_p}$ such that $r_0^p = [0,1)^2$, $r_n^p = r$ and $r_k^p \in \texttt{qsplit}(r_{k-1}^p)$ for all $1 \le k \le n_p$. Let $(p_i)_{1 \le i \le |\mathcal{P}|}$ be an enumeration of the elements of $\mathcal{P}$. The sequence $(r_k^{p_i})_{0 \le k < n_{p_i}, 1 \le i \le |\mathcal{P}|}$ induces a quad-tree (V_N, E_N) with $\mathcal{P} \subset V_N$. The

fact that $r_k \in V_k$ for all $0 \leq k < N$ is immediate. We show that $\mathcal{P}$ is the set of terminal nodes $L(V_N)$ of (V_N, E_N). Assume there is some $q \in \mathcal{P}$ such that $q \notin L(V_N)$. Then the quad split $\texttt{qsplit}(q)$ is a subset of V_N, i.e. $q = r_k^p$ for some $p \in \mathcal{P}$ and $0 \leq k < n_p$, but then $q \subsetneq p$ for $q, p \in \mathcal{P}$, which is a contradiction. It remains to show that each $p \in L(V_N)$ is also element of $\mathcal{P}$. Because $\mathcal{P}$ and $L(V_N)$ are partitions, $\mathcal{P} \subsetneq L(V_N)$ is impossible, therefore $\mathcal{P} = L(V_N)$. □

Lemma 2.2.7 and Lemma 2.2.9 give a supplementary justification for the notion of 'depth' in Definition 2.2.1. The depth of a dyadic partition is the maximal number of edges leading from the root S to the terminal nodes of the associated quad-tree. The Lemmas add up to the following important result:

Theorem 2.2.10: *The class of dyadic partitions of* $[0,1)^2$ *is the class of quad-tree partitions of* $[0,1)^2$.

Theorem 2.2.10 is a meaningful characterization of the set of dyadic partitions. It shows that dyadic partitions have a tree structure that can be utilized in the development of algorithms: Traversing the set of quad-trees is obviously equivalent to traversing the set of dyadic partitions. Before we provide an algorithm, we need to comment on dyadic partitions on a discrete image domain.

2.2.2 Discrete Image Domain

To be able to define dyadic partitions on a discrete index set, let the image domain now be given by a square $S = \{0, \ldots, 2^N - 1\}^2 = [0, 2^N)^2 \cap \mathbb{Z}^2$, $N \in \mathbb{N}$. We take the interval bounds 0 and $2^N - 1$ instead of 1 and 2^N, because this allows us to carry over the notions dyadic square and quad split from the continuous case in a natural way.

Definition 2.2.11 (Dyadic Square, Dyadic Partition): *Consider a discrete image domain* $S = \{0, \ldots, 2^N - 1\}^2$ *and for each* $0 \leq n \leq N$ *let the following set be given:*

$$\mathcal{D}_n = \{[(i-1)2^n, i2^n) \times [(j-1)2^n, j2^n) \cap \mathbb{Z}^2 : 1 \leq i, j \leq 2^{N-n}\}$$

Each element of $\mathcal{D} = \bigcup_{n=0}^{N} \mathcal{D}_n$ *will be called a* **(discrete) dyadic square**.

A partition $\mathcal{P}$ of S with $\mathcal{P} \subset \mathcal{D}$ will be called a **(discrete) dyadic partition**.

We will omit the word 'discrete' when a discrete context is obvious.

At first glance, Definition 2.2.11 provides a canonical extension of the partitions introduced in Definition 2.2.1 to an image domain larger than $[0,1)^2$. Nevertheless, the redefinition of partitions and dyadic square for the discrete image domain could not be avoided. The following observation establishes another link between continuous and discrete partitions.

Lemma 2.2.12: *Let $S = [0, 2^N)^2 \cap \mathbb{Z}^2$, and let a dyadic partition $\mathcal{P}$ of $[0,1)^2$ with a depth less than or equal to $N \in \mathbb{N}$ be given. Then the discretization $\boldsymbol{\delta}^S(\mathcal{P})$ is a discrete dyadic partition of S. Moreover, for each dyadic partition $\mathcal{Q}$ of S there is one and only one dyadic partition $\mathcal{P}$ of $[0,1)^2$ such that $\mathcal{Q} = \boldsymbol{\delta}^S(\mathcal{P})$.*

Proof. By Lemma 2.1.4, the discretization $\boldsymbol{\delta}^S$ transforms a partition of $[0,1)^2$ consisting of rectangles into a partition of S. Now consider a dyadic square r in $[0,1)^2$,

$$r = [(i-1)2^{-k}, i2^{-k}) \times [(j-1)2^{-k}, j2^{-k}),\ 1 \le i, j \le 2^k, 0 \le k \le N.$$

Then

$$\begin{aligned}\boldsymbol{\delta}^S(r) &= [(i-1)2^{N-k} - \frac{1}{2}, i2^{N_k} - \frac{1}{2}) \times [(j-1)2^{N-k} - \frac{1}{2}, j2^{N-k} - \frac{1}{2}) \cap \mathbb{Z}^2 \\ &= [(i-1)2^{N-k}, i2^{N_k}) \times [(j-1)2^{N-k}, j2^{N-k}) \cap \mathbb{Z}^2\end{aligned}$$

is a dyadic square. Thus, if $\boldsymbol{\delta}$ is a dyadic partition of $[0,1)^2$, then $\boldsymbol{\delta}^S(\mathcal{P})$ is a dyadic partition of S. Let now a dyadic partition $\mathcal{Q}$ of S be given. For each $q \in \mathcal{Q}$ there are numbers $0 \le k_q \le N$ and $1 \le i_q, j_q \le 2_q^k$ such that

$$r' = [(i_q - 1)2^{N-k_q}, i_q 2^{N-k_q}) \times [(j_q - 1)2^{N-k_q}, j_q 2^{N-k_q}) \cap \mathbb{Z}^2.$$

Let

$$r_q = 2^{-N} r' = [(i_q - 1)2^{-k_q}, i_q 2^{-k_q}) \times [(j_q - 1)2^{-k_q}, j_q 2^{-k_q})$$

for all $q \in \mathcal{Q}$. Then $q = \boldsymbol{\delta}^S(r_q)$ for all $q \in \mathcal{Q}$ and $\mathcal{P} = \{r_q : q \in \mathcal{Q}\}$ is a dyadic partition of $[0,1)^2$. Let r and r' be dyadic squares with $r \neq r'$. Then $\boldsymbol{\delta}^S(r) \neq \boldsymbol{\delta}^S(r')$. Therefore, $\mathcal{P}$ is unique. □

By virtue of the link between continuous and discrete partitions given in the previous Lemma, discrete dyadic partitions also have a quad-tree structure. We do not repeat all the results from the previous section. For the discrete case we only redefine the notion quad split, and show that it is one to one with the continuous case.

Definition 2.2.13: *Let r be a dyadic square in S with side length $2^n > 1$,$n \in \mathbb{N}$, $r = [(i-1)2^n, i2^n) \times [(j-1)2^n, j2^n) \cap \mathbb{Z}^2$ $(1 \leq i \leq j \leq n)$, and let (m_1, m_2) be the 'midpoint' ($m_1 = (2i-1)2^{n-1}$, $m_2 = (2j-1)2^{n-1}$). A* **(discrete) quad split** *of r is the following set of rectangles:*

$$\texttt{qsplit}(r) = \{[s_1, m_1) \cap \mathbb{Z}, [m_1, t_1) \cap \mathbb{Z}\} \boxtimes \{[s_2, m_2) \cap \mathbb{Z}, [m_2, t_2) \cap \mathbb{Z}\}.$$

For a square r with side length 1 we define $\texttt{qsplit}(r) = \emptyset$.

Note that the quad split of a dyadic square consists of four dyadic squares with side lengths 2^{n-1}.

Lemma 2.2.14: *Let $S = [0, 2^N)^2 \cap \mathbb{Z}^2$, and let r be a continuous dyadic square with side length greater than or equal to 2^{-N}. Then $\boldsymbol{\delta}^S(\texttt{qsplit}(r)) = \texttt{qsplit}(\boldsymbol{\delta}^S(r))$.*

Proof. We adopt the notation used in proof of Lemma 2.2.12: $\boldsymbol{\delta}(r) = 2^N r \cap \mathbb{Z}^2$. The result follows due to the analogy of Definition 2.2.13 and Definition 2.2.2. □

It would now be possible to redefine the notions quad-tree and quad-tree partition for the discrete case, and to give exactly the same results as already developed in the previous section for the continuous case. Instead, we argue by analogy and point out that discrete dyadic partitions have a recursive (quad) tree structure with a production rule, as in the continuous case. Using this analogy we repeat the statement of Theorem 2.2.10 for the discrete case in the following:

Theorem 2.2.15: *Let S be dyadic, $S = [0, 2^N)^2 \cap \mathbb{Z}^2$. For each dyadic partition $\mathcal{P}$ of S there is one and only one quad-tree inducing $\mathcal{P}$.*

Remark 2.2.16: *If the (discrete) image domain is not a dyadic square, then a dyadic partition does not exist. Nevertheless, a quad-tree partition can also be defined with a quad split on* **rectangles** *on S. For a rectangle $r = \{s_t, \dots, t_1\} \times \{s_2, \dots, t_2\}$, a quad split is defined as the discretization of $\boldsymbol{\delta}^S(\texttt{qsplit}([s_1, t_1) \times [s_2, t_2))) \setminus \emptyset$. By the rounding involved in the quad split process, the rectangles do have different sizes, in general. This leads to the following effects: First, the quad split may (depending on the stopping criteria in the tree) consist of only two rectangles (see Figure 2.6). And second, the partition induced by the quad-tree is, in general, not equivalent to the partition induced with the 'same' quad-tree on a continuous domain.*

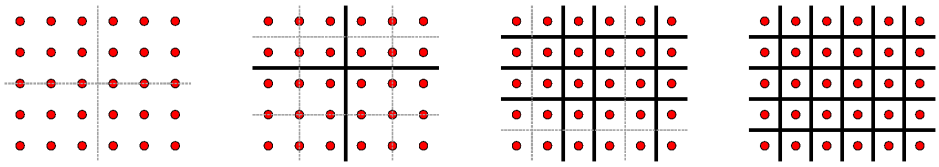


Fig. 2.6: *Recursive quad splits of a discrete rectangle*

2.2.3 Recursion

The tree structure of (discrete) dyadic partitions will now be exploited for developing an efficient algorithm that traverses the family of dyadic partitions for the purpose of minimizing the Potts functional. By Theorem 2.2.15, a dyadic partition of S is one to one with a quad-tree.

Before we can show how the recursive structure of the partitions can be used to minimize the Potts functional, we need some definitions. Consider a finite set Q and a family of sets $\{A_q : q \in Q\}$. We define

$$\bigotimes_{q \in Q} A_q := \{\bigcup_{q \in Q} a_q : a_q \in A_q \ \forall \ q \in Q\}.$$

Now for a dyadic square r let the following recursion formula

$$\mathfrak{Q}_r = \{\{r\}, \bigotimes_{q \in \texttt{qsplit}(r)} \mathfrak{Q}_q\}$$

be given. Then, according to Definition 2.2.4, the set of quad-tree partitions of S is $\mathfrak{Q}_S$. Moreover, for each dyadic square r, assume $h_r^* := \inf_{f_r \in \mathcal{F}_r} \|f_r - z|_r\|_2^2$. Then, with the definition $K_{\mathcal{P}} = \gamma|\mathcal{P}| + \sum_{s \in \mathcal{P}} h_s^*$ for fixed $\gamma \in \mathbb{R}$ and data z, the following holds

$$\inf_{(\mathcal{P}, f_{\mathcal{P}}) \in \mathfrak{S}} H_\gamma(z, (\mathcal{P}, f_{\mathcal{P}})) = \min_{\mathcal{P} \in \mathfrak{Q}_S} K_{\mathcal{P}}.$$

Assume that $\mathcal{Q} = \texttt{qsplit}(r)$, and consider for each $q \in \mathcal{Q}$ a partition $\mathcal{P}_q$ of q, then

$$K_{\bigcup_{q\in\mathcal{Q}}\mathcal{P}_q} = \gamma|\bigcup_{q\in\mathcal{Q}}\mathcal{P}_q| + \sum_{p\in\bigcup_{q\in\mathcal{Q}}\mathcal{P}_q} h_p^* = \sum_{q\in\mathcal{Q}}\gamma|\mathcal{P}_q| + \sum_{q\in\mathcal{Q}}\sum_{r\in\mathcal{P}_q} h_r^* = \sum_{q\in\mathcal{Q}} K_{\mathcal{P}_q}.$$

This leads to the recursion rule

$$\min_{\mathcal{P}\in\mathfrak{Q}_r} K_{\mathcal{P}} = \begin{cases} \min\{\gamma + h_r^*, \sum_{q\in\texttt{qsplit}(r)} \min_{\mathcal{P}\in\mathfrak{Q}_q} K_{\mathcal{P}}\}, & \text{if } \texttt{qsplit}(r) \neq \emptyset, \\ \gamma + h_r^* & \text{otherwise.} \end{cases} \tag{2.6}$$

used for the minimization algorithm.

Remark 2.2.17: *Crucial in the development of recursion equation (2.6) was the fact that the Potts functional fulfills the reduction principle. But a little bit more was needed: If a functional fulfills the reduction principle, then it can be written in the form $\varphi_{\mathcal{P}}((h_r(f_r))_{r\in\mathcal{P}})$. For the breakup of the functional into functionals on sub-partitions of $\mathcal{P}$ a kind of separability of $\varphi_{\mathcal{P}}$ is additionally necessary. In the case of the Potts functional this is $\varphi_{\mathcal{P}\cup\mathcal{Q}} = \varphi_{\mathcal{P}} + \varphi_{\mathcal{Q}}$.*

The algorithm for traversing all quad-trees of S will consist of four steps. First, the definition and creation of an abstract tree object for storage of projection results. Second, a traversal for doing the local minimization at each node. The third step is the minimization of the Potts functional using this resulting tree. In the last step, the projections are performed on the minimizing quad-tree.

As in Subsection 2.1.3, we use some abstract data types for the passing of parameters in the algorithms. Again we make use of an abstract reference object REF of generic type OBJECT for accessing global data within procedures. For the representation of the quad-tree we introduce a dynamic object NODE of abstract type NodeOBJECT. A NODE contains references to four subnodes viz. NODE.ll, NODE.lr, NODE.ul and NODE.ur, the corresponding rectangle NODE.$r \subset S$, and more entries for the data. If a node NODE is empty then we denote this by NODE = NIL. The creation of a node will be denoted by NEW(NODE).

The following procedures and the algorithm are designed such that a quad-tree is applicable to a rectangular finite image domain $S = \{0,\ldots,N_1-1\} \times \{0,\ldots,N_2-1\}$, which does not have to be a dyadic square, see Remark 2.2.16. Applied to a dyadic square, however, they correspond to dyadic partitions.

We start with the creation of the quad-tree. A call of `Create`(x, y, w, h) creates an empty quad-tree over the rectangle $\{x, \dots, x+w\} \times \{y, \dots, y+h\}$ $(y, x, w, h \in \mathbb{N})$.

Procedure `CreateTree`(*x,y,w,h: INTEGER*): NodeOBJECT

input : geometry of rectangle x, y, w, h
output: instance of node NODE and subtree if geometry permits that, NIL otherwise

```
begin
    if h > 0 and w > 0 then
        NEW(NODE);
        NODE.r ← [x, x + w) × [y, y + h) ∩ ℤ²;
        w₂ ← ⌊w/2⌋; h₂ ← ⌊h/2⌋;
        NODE.ll ← CreateTree(x, y, w₂, h₂);
        NODE.lr ← CreateTree(x + w₂, y, w − w₂, h₂);
        NODE.ul ← CreateTree(x, y + h₂, w₂, h − h₂);
        NODE.ur ← CreateTree(x + w₂, y + h₂, w − w₂, h − h₂);
    else
        NODE ← NIL
    end
    return NODE;
end
```

Given a node NODE, the following procedure computes the local minima for the complete subtree attached to NODE.

Procedure `LocalMin`(NODE*: NodeOBJECT;* REF*:OBJECT*)

input : NODE and data z stored in REF
output: the local projection associated to NODE.r stored in NODE.h

```
begin
    r ← NODE.r;
    if NODE ≠ NIL then
        NODE.h ← ‖Π_{F_r} z − z_r‖²;
        foreach snode ∈ {NODE.ll, NODE.lr, NODE.ul, NODE.ur} do
            LocalMin (snode,REF)
        end
    end
end
```

Procedure `MinTree` minimizes the Potts functional, and accordingly sets the nodes to status 'isterminal=TRUE' or 'isterminal=FALSE'. The return value is the minimizer of the Potts functional restricted to the domain associated to the given node.

Procedure `MinTree`(NODE*: NodeOBJECT;* $\gamma \in \mathbb{R}$): REAL

```
input  : NODE and parameter γ ∈ ℝ
output: Potts functional minimum, minizing partition by setting tag NODE.isterminal to TRUE
        or FALSE
begin
    r ← NODE.r;
    if NODE ≠ NIL then
        this ← NODE.h + γ;
        subnodepresent ← FALSE; sub ← 0;
        foreach snode ∈ {NODE.ll, NODE.lr, NODE.ul, NODE.ur} \ NIL do
            subnodepresent ← TRUE;
            sub ← sub + MinTree(snode, γ);
        end
        if subnodepresent= FALSE then
            sub ← ∞
        end
        if this ≤ sub then
            NODE.isterminal ← TRUE;
            return this
        else
            NODE.isterminal ← FALSE;
            return sub
        end
    else
        return ∞
    end
end
```

Procedure `TreeProjection` performs the local projections after the global minimization is done.

Procedure `TreeProjection`(NODE*:NodeOBJECT;* REF*:OBJECT*)

```
input  : NODE and parameter γ ∈ ℝ
output: Minimizer of the Potts functional in REF.f
begin
    r ← NODE.r; z ← REF.z;
    if (NODE.isterminal) then
        REF.f|_r ← Π_{F_r} z;
    else
        foreach snode ∈ {NODE.ll, NODE.lr, NODE.ul, NODE.ur} \ NIL do
            TreeProjection (snode,REF)
        end
    end
end
```

The four components, tree creation, local minimization, global minimization and projection have been arranged and we can now formulate an algorithm for minimizing the Potts functional over dyadic partitions.

Algorithm 2.8: Minimizing the Potts Functional over Dyadic Partitions.

input : Data z over image domain $S = \{0, \dots, N_1 - 1\} \times \{0, \dots, N_2 - 1\}$, parameter $\gamma \in \mathbb{R}$
output: Minimizer f

begin
  REF.$z \leftarrow z$;
  NODE $\leftarrow$ `CreateTree`$(0, 0, N_1, N_2)$;
  `LocalMin`(NODE, REF);
  $min \leftarrow$ `MinTree`(NODE, γ);
  `TreeProjection`(NODE, REF);
  $f \leftarrow$ REF.f;
end

For Algorithm 2.8, the following complexity result holds:

Lemma 2.2.18: *Let $S = \{0, \dots, N_1-1\} \times \{0, \dots, N_2-1\}$. Assume for a rectangle r in S with side lengths w and h that the complexity of the computation of $\|\Pi_{\mathcal{F}_r} z - z_r\|_2^2$ is given by $C_1(w, h)$, and let $C_2(|r|)$ be the complexity for the projection operation $\hat{f}_r \leftarrow \Pi_{\mathcal{F}_r} z$. Let $d = \min\{\lfloor \log_2(N_1) \rfloor, \lfloor \log_2(N_2) \rfloor\}$. Then the complexity of Algorithm 2.8 is given by*

$$O\left(\sum_{i=0}^{d} 4^i C_1\left(\frac{N_1}{2^i}, \frac{N_2}{2^i}\right) + \sum_{r \in \hat{\mathcal{P}}} C_2(|r|) \right).$$

Proof. Let $C(d)$ be the complexity of recursion step d. Let $C_{loc}(d)$ be the complexity of local operations in step d. Then the recursion formula $C(k) = C_{loc}(k) + 4C(k+1)$, $0 \le k \le d$ leads to $C(d) = \sum_{k=0}^{d} 4^k \cdot C_{loc}(k)$. The complexities of node creation and comparisons in the minimization and reconstruction procedures are assumed to be constant. In recursion step k the side lengths of the rectangles are given by $w = \frac{N_1}{2^k}$ and $h = \frac{N_2}{2^k}$. In the projection step operations on sets not in $\hat{\mathcal{P}}$ have constant complexity. Setting these values in the recursion formula yields the result. □

Lemma 2.2.18 indicates that the complexity of the minimization process is crucially dependent on the efficiency of the local projections. The complexity of Algorithm 2.8 for projections with maximal efficiency is given in the following

Corollary 2.2.19: *Let the assumptions of the previous Lemma be fulfilled. Assume that for some constant $c_1 > 0$, which may depend on other variables than the size, the local minimization complexity is given by $C_1(w, h) =$*

$c_1(w+h)$ and that the local reconstruction complexity $C_2(\cdot)$ is linear. Then Algorithm 2.8 has complexity $O(c_1|S|)$.

Proof. Let $C_2(m) = m \cdot c_2$ for all $m \in \mathbb{N}$, then

$$\begin{aligned}\sum_{i=0}^{d} 4^i C_1(\frac{N_1}{2^d}, \frac{N_2}{4^d}) + \sum_{r\in\hat{\mathcal{P}}} C_2(|r|) &= \sum_{i=0}^{d} c_1 \cdot 2^i(N_1+N_2) + \sum_{r\in\hat{\mathcal{P}}} C_2(|r|)\\ &= c_1 \cdot (N_1+N_2)\cdot(2^{d+1}-1) + c_2|S| \le (c_1+c_2)|S|\\ &\le c_1 \cdot (N_1+N_2)\cdot 2\min\{N_1,N_2\} + c_2|S|\\ &\le c_1 \cdot 4N_1 \cdot N_2 + c_2|S| = (4c_1+c_2)|S|.\end{aligned}$$

□

2.3 Synopsis

In this chapter we have introduced two classes of partitions: hierarchic and dyadic partitions. Hierarchic partitions have a recursive structure in the dimension, while dyadic partitions are one to one with quad-trees and thus have a tree structure. For both partition types the discrete and continuous cases have to be treated separately, but a canonical discretization provides a useful connection between these cases.

We have provided algorithms that traverse the set of partitions for the purpose of minimizing the Potts functional. In an optimal case, when the local approximations can be done in constant time and the projections can be performed with linear complexity, the Potts functional can be minimized with complexity $O(|S|^2)$ for hierarchic partitions, while for dyadic partitions the minimization complexity is $O(|S|)$. In the next chapter, we will deal with efficient local minimizations and projections.

The recursion rules and resulting algorithms have been developed for the minimization of the Potts functional. However, with some more notational overhead it is possible to do that for quite a large class of functionals. A necessary condition is that the functionals fulfill the reduction principle. Another condition is a sort of separability of the function $\varphi_{\mathcal{P}}$ associated to a functional f.

There are other kinds of partitions allowing a manageable recursion. We cite as an example the tree serial dynamic programming approach originating

from V. Mottl, see Mottl and Muchnik (2002) and Mottl et al. (1998). Moreover both dyadic and hierarchic partitions may be refined in various ways. For example, each rectangle may additionally be divided into two parts with a straight line resulting in two *wedges*. This refinement can be formulated as a property of the partition, but it can also be formulated as an additional feature of the local regression model. We follow up with the second approach because we want to keep a clear structure of the hierarchic and dyadic partitions. We prefer to treat the notationally expensive description of non-rectangular divisions of the fragments in a separate section.

3. Local Models

In the previous chapter we have introduced two different approaches to partition a two-dimensional image domain, both of them allowing an efficient enumeration of the partition class. We have given algorithms for the minimization of the Potts functional for an arbitrary class of admissible functions. The efficiency of these algorithms crucially depends on the complexity of the local projections on the space of admissible functions over rectangles, compare Lemma 2.1.7 and Lemma 2.2.18.
This chapter is about local models over a domain that is not necessarily rectangular. In Section 3.1 we make some general remarks concerning the minimization of certain functionals such as existence and uniqueness of minimizers. In Section 3.2 we comment on wedge intersection, the division of the image domain into two parts by a line. In Section 3.3 we present a highly efficient scheme to compute local minimizers and projections on a polygonal domain. This rule will permit a fast minimization of the Potts functional.

3.1 Local Regression

Let the image domain be finite, $S = \{1, \dots, N_1\} \times \{1, \dots, N_2\}$, $N_1, N_2 \in \mathbb{N}$. Let $\mathcal{R}$ be the set of connected subsets of S, and let $\mathcal{F} = (\mathcal{F}_r)_{r \in \mathcal{R}}$ be a class of admissible functions. Assume that data $z \in \mathbb{R}^S$ is observed.

With $z_r := z|_r$ and $\varrho : \mathbb{R} \to \mathbb{R}$, $\varrho(x) = x^2$ for all $r \in \mathcal{R}$, the Potts functional has the form

$$H_{\gamma,z}(\mathcal{P}, f_{\mathcal{P}}) = \gamma|\mathcal{P}| + \sum_{r \in \mathcal{P}} \sum_{i \in r} \varrho(z_i - f_{\mathcal{P}}(i)).$$

This thesis is mainly about the minimization of a Potts functional with L^2 distance, i.e. $\varrho(x) = x^2$. Nevertheless, we want to comment on the minimization of

$$h_r(f_r, z) := \sum_{i \in r} \varrho(z_i - f_r(i))$$

with a general choice of the function ϱ. Interesting examples are robust functions such as the so called Hampel cup function, see Winkler (2002),

pp. 31-41 and Winkler and Liebscher (2002). For the one-dimensional case efficient algorithms for the minimization of $H_{\gamma,z}$ with a replacement of $\varrho(\cdot) = \|\cdot\|_2^2$ by some robust functions have already been developed, see Brandt and Hutzenthaler (2004). There may be ways to extend the results that will be presented in Section 3.3 from the L^2 distance to more robust functions.

In this section we consider the minimization problem

$$h_r(f_r, z) \xrightarrow{!} \min_{f_r \in \mathcal{F}_r} h_r(f_r, z). \tag{3.1}$$

3.1.1 Existence of Minimizers

In this paragraph sufficient conditions on ϱ and the class of admissible functions $\mathcal{F}$ for the existence of solutions of the minimization problem (3.1) are given.

We start with two definitions: A function $F : \mathbb{R}^n \to \bar{\mathbb{R}}$ is called **level bounded**, if the set $\operatorname{lev}_{\leq\alpha} F := \{x \in \mathbb{R}^n : F(x) \leq \alpha\}$ is bounded or empty for all $\alpha \in \mathbb{R}$. A function $G : \mathbb{R}^n \to \bar{\mathbb{R}}$ is called **proper**, if the set $\{x \in \mathbb{R}^n : G(x) < \infty\}$ is nonempty and closed.

Lemma 3.1.1: *Let $\varrho : \mathbb{R} \to \mathbb{R}$ be a positive, lower semicontinuous and level bounded function. Assume $\mathcal{F}_r = \mathbb{R}^r$ or $\mathcal{F}$ is a closed subset of $\mathbb{R}^r$. Then the set*

$$\operatorname*{argmin}_{f_r \in \mathcal{F}_r} \sum_{i \in r} \varrho(z_i - f_r(i))$$

is nonempty and compact.

Proof. By Theorem 1.9 in Rockafellar and Wets (2004), p.11, the set of minima of a lower semicontinuous level-bounded proper function $g : \mathbb{R} \to \bar{\mathbb{R}}$ is nonempty and compact. Thus, it has only to be proved that the function $f_r \mapsto \sum_{i \in r} \varrho(f_r - z_i)$ is lower semicontinuous and level bounded. Now consider the two positive, symmetric, lower semicontinuous functions $g_1, g_2 : \mathbb{R} \to \mathbb{R}$. Then the sum $g_1 + g_2 : (x_1, x_2) \mapsto g_1(x_1) + g_2(x_2)$ is also symmetric, positive and lower semicontinuous. Moreover, for all $\alpha > 0$ the set $\{(x_1, x_2) : g_1(x_1) + g_2(x_2) \leq \alpha\}$ is bounded because by the positivity of g_1 and g_2, it is contained in the set $\{(x_1, x_2) : g_1(x_1) \leq \alpha, g_2(x_2) \leq \alpha\}$ which is bounded. For the existence of a minimum on a closed subset $\mathcal{F}_r$ of $\mathbb{R}^r$, consider the minimization of $f_r \mapsto \bar{h}_r(f_r)$ with the proper function $\bar{h}_r : \mathbb{R}^r \to \bar{\mathbb{R}}$ defined

by

$$\bar{h}_r(f_r) := \begin{cases} h_r(f_r) \text{ if } f_r \in \mathcal{F}_r, \\ \infty \text{ otherwise.} \end{cases}$$

Note that level boundedness is hereditary: The set $\mathrm{lev}_{\leq\alpha} h_r|_{\mathcal{F}_r}$ is trivially empty or bounded for each α if the same is true for the whole space $\mathbb{R}^r$. □

Conditions of Lemma 3.1.1 are only sufficient; they are not necessary conditions for existence of minima. Consider, for example, $\rho \equiv 1$. Nevertheless, none of them can be omitted in the Lemma. This will be demonstrated with the following three counter-examples.

Example 3.1.2: *Consider a fragment r with two elements, $r = \{1, 2\}$, constant data $z_1 = z_2$, quadratic distance $\varrho(x) = x^2$ and the (open) set of non constant functions $\mathcal{F}_r := \{f : \{1, 2\} \to \mathbb{R} : f(1) \neq f(2)\}$. Then the set $\operatorname{argmin}_{f_r \in \mathcal{F}_r} h_r(f_r, z)$ is empty since h_r attains it unconstrained minimum at $f(1) = f(2) = z_1$ on the boundary of the open set $\mathcal{F}_r$.*

The lower semicontinuouity of ϱ is also crucial:

Example 3.1.3: *Consider the (upper semicontinuous) function*

$$\varrho(x) = \begin{cases} |x| & \text{if } |x| < 1, \\ 2|x| & \text{otherwise,} \end{cases}$$

data $z = \{-1, 1\}$ over $r := \{1, 2\}$ and constant regression $\mathcal{F}_r = \{f : \{1, 2\} \to \mathbb{R} : f(1) = f(2) = \mu, \mu \in \mathbb{R}\}$.

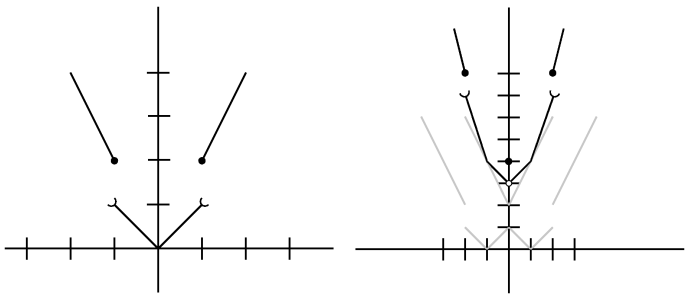


Fig. 3.1: *ϱ and $h_r(\cdot, z_r)$*

Then

$$h_r(\mu, z_r) = \begin{cases} 4|\mu| & \text{if } |\mu| > 2, \\ 2 + 2|\mu| & \text{if } 1 \leq |\mu| < 2, \\ 3 + |\mu| & \text{if } 0 < |\mu| < 1, \\ 4 & \text{if } x = 0, \end{cases}$$

which does not attain its infimum.

Example 3.1.4: *That* $\mathrm{lev}_{\leq\alpha}\, \varrho$ *is bounded for* all $\alpha \in \mathbb{R}$ *is a necessary condition in Lemma 3.1.1 and cannot be replaced by the weaker requirement of* existence *of some* $\alpha \in \mathbb{R}$ *such that* $\mathrm{lev}_{\leq\alpha}\, \varrho$ *is bounded (even though* $\mathrm{lev}_{\leq\alpha}$ *is compact). Consider the function*

$$\varrho(x) = \begin{cases} 1 + \frac{1}{|x|} & \text{if } |x| > 1, \\ 2(2x+1)^2 & \text{if } -1 \leq x \leq 0, \\ 4 - 2(2x-1)^2 & \text{if } 0 < x \leq 1, \end{cases}$$

which has a compact lower level set $\mathrm{lev}_{\leq\alpha}\, \varrho$ *for all* $0 \leq \alpha < 1$ *and empty lower level sets for* $\alpha < 0$*. Assume data* $z = \{0, 0\}$ *over* $r = \{1, 2\}$ *and the set of antisymmetric functions* $\mathcal{F}_r = \{f : \{1, 2\} \to \mathbb{R} : f(1) = -f(2) = \mu\}$*.*

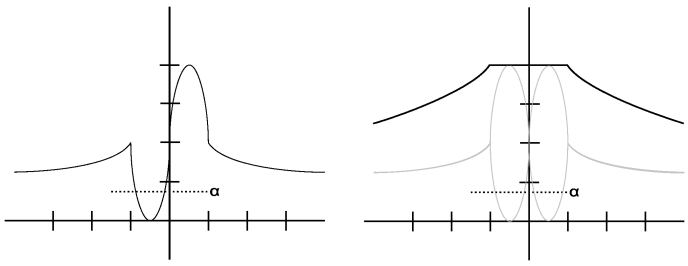


Fig. 3.2: ϱ *and* $h_r(\cdot, z_r)$

Then

$$h_r(\mu, z_r) = \begin{cases} 2 + \frac{2}{|\mu|} & \text{if } |\mu| > 1, \\ 4 & \text{otherwise}, \end{cases}$$

with an infimum at infinity.

The previous example shows an important fact about the existence of minima, which is stated in the following

Remark 3.1.5: *Level boundedness is hereditary, i.e. the restrictions of level bounded functions to closed subsets, in particular to subspaces, are level bounded. In the following we will call the property that there is one $\alpha \in \mathbb{R}$ such that $\mathrm{lev}_{\leq\alpha}\, \varrho$ is bounded* **local level boundedness**. *The crucial point is that local level boundedness is not hereditary, i.e. it does not imply local level boundedness on closed subsets.*

Nevertheless, if there is no restriction on the function space $\mathcal{F}_r$ for each $r \in \mathcal{R}$, local level boundedness of ϱ implies local level boundedness of h_r. This is due to the fact that

$$\begin{aligned}\{x \in \mathbb{R}^r : \sum_{i\in r} \varrho(x_i - z_i) \leq \alpha\} &\subset \prod_{i\in r} \{x_i \in \mathbb{R} : \varrho(x_i - z_i) \leq \alpha\} \\ &= \prod_{i\in r} (\{x_i : \varrho(x_i) \leq \alpha\} + z_i)\end{aligned}$$

which is a bounded set. Because in most cases we assume that $\varrho(\cdot)$ has a minimum at $x = 0$, this result is not particularly exciting: Minimizing h_r over $\mathcal{F}_r = \mathbb{R}^r$ would result in data z: $f_r(i) = z_i$ for all $i \in r$ and all $r \in \mathcal{R}$.

The reason why local level boundedness of ϱ does not necessarily imply local level boundedness of h_r over subspaces of $\mathbb{R}^r$ is that the local (existing!) bounded level set of h_r does not have to intersect a given subspace at all, see the following figure.

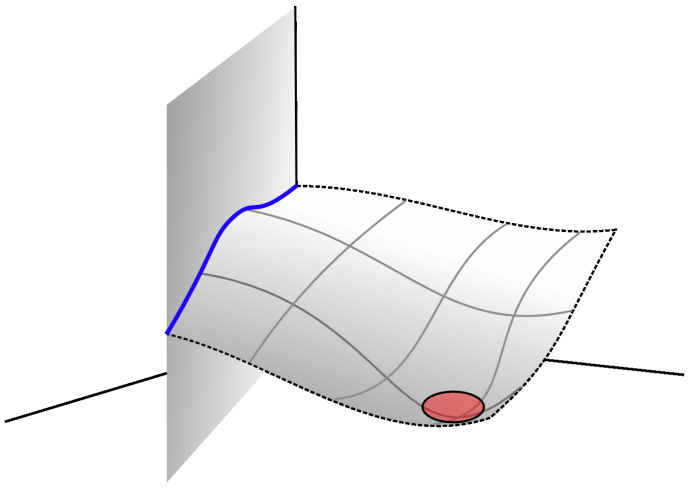


Fig. 3.3: *Local level boundedness does not imply level boundedness on subspaces.*

As a consequence, for robust distances such as the cup shaped Huber func-

tion, which does not satisfy the assumption of (global) level boundedness, we have to find other criteria or – in the worst case – examine them bit-by-bit.

As a direct consequence of Lemma 3.1.1 we can state the following:

Corollary 3.1.6: *Let* $h_r(f_r, z) := \sum_{i \in r}(f_r(i) - z_i)^2$. *Assume* $\mathcal{F}_r = \mathbb{R}^r$ *or* $\mathcal{F}_r$ *is a closed subset of* $\mathbb{R}^r$. *Then the set* $\operatorname{argmin}_{f_r \in \mathcal{F}_r} h_r$ *is nonempty and compact.*

Proof. The function $\varrho : \mathbb{R} \to \mathbb{R}$, $\varrho(x) = x^2$, is positive, lower semicountinuous and level bounded. Apply Lemma 3.1.1. □

3.1.2 Uniqueness of Minimizers

We still consider the minimization problem (3.1)

$$h_r(f_r, z) \xrightarrow{!} \min_{f_r \in \mathcal{F}_r} h_r(f_r, z)$$

with $h_r(f_r, z) = \sum_{i \in r} \varrho(f_r(i) - z_i)$ and give a sufficient condition for the uniqueness of a minimizer of h_r.

We cite the following theorem from Hirzebruch and Scharlau (1996), p.74. Recall that a space X is called strictly normed, if for all $x, z \in X \setminus 0$ with $\|x + z\| = \|x\| + \|z\|$ there is some $\lambda > 0$ such that $x = \lambda z$.

Theorem 3.1.7 (Approximation Theorem): *Let* X *be a real, normed space. Let* $a \in X$, *and let* W *be a convex, closed subset of* X.
If X *is strictly normed, then there is at most one* $x \in W$ *such that*

$$\|a - x\| = \inf_{z \in W} \|a - z\|. \tag{3.2}$$

Remark 3.1.8: *Hirzebruch and Scharlau (1996) give a class of functions, for which the approximation problem 3.2 has a unique solution. These are the so called uniformly convex Banach spaces, thereunder the spaces* $L^p(\mathbb{R}^n, \varphi)$ *for all* $1 < p < \infty$. *See Hirzebruch and Scharlau (1996), pp. 74-75.*

From Theorem 3.1.7 we deduce the following important result:

Theorem 3.1.9: *Let $\varrho = |x|^p$, $1 < p < \infty$, and let $\mathcal{F}_r$ be (a subspace of) $\mathbb{R}^r$. Then the minimization problem $h_r(\cdot, z) \stackrel{!}{\longrightarrow} \operatorname{argmin}_{f_r \in \mathcal{F}_r} \sum_{i \in r} \varrho(f_r - z_i)$ has a unique solution.*

Proof. Observe that $\operatorname{argmin}_{f_r \in \mathcal{F}(r)} \sum_{i \in r} |z_i - f_r(i)|^p = \operatorname{argmin}(\sum_{i \in r} |z_i - f_r(i)|^p)^{1/p} = \operatorname{argmin} ||z_r - f_r||_p$. Subspaces are closed and convex. Minkowski's Inequality for Sums states that if $p > 1$ and $x_k, z_k > 0$ for all k then $\|x + z\|_p \leq \|x\| + \|z\|$ with equality if and only if there is some $\lambda > 0$ such that $x = \lambda z$, see Abramovitz and Stegun (1972), p. 11. □

As a counter-example for the not uniformly convex space L^2, consider the following

Example 3.1.10: *Let $\varrho(x) = |x|$ corresponding to the L^1 distance h_r. Let data $z = \{-1, 1\}$ over $r = \{1, 2\}$ be given, and consider constant regression $\mathcal{F}_r := \{f : \{1, 2\} \to \mathbb{R} : f(1) = f(2) = \mu, \mu \in \mathbb{R}\}$.*

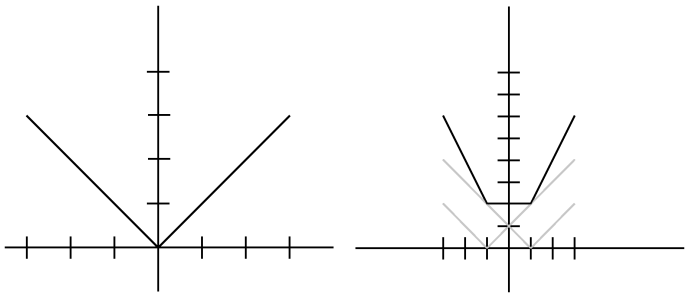


Fig. 3.4: *$\varrho = \| \cdot \|_{L^1}$ and $h_r(\cdot, z_r)$*

Then

$$h_r(\mu, x) = \begin{cases} 2|\mu| & \text{if } |\mu| > 1, \\ 2 & \text{otherwise.} \end{cases}$$

Thus h_r does not have a unique minimum. Note that the minimizers are the median values of the data z_r.

3.1.3 Least Squares Regression

We consider a finite subset r of $S \subset \mathbb{Z}^2$ and data $z \in \mathbb{R}^r$. Now we focus on the minimization of the data term $h_r(f_r, z) = \|f_r - z|_r\|_2^2$ of the Potts

functional. Applied to this least squares regression problem, Theorem 3.1.9 immediately leads to the following

Corollary 3.1.11: *Let $\mathcal{F}_r$ be (a subspace of) $\mathbb{R}^r$. Then for each $z \in \mathbb{R}^r$ the functional $h_r(f_r, z) = \sum_{i \in r}(f_r(i) - z_i)^2$ has a unique minimizer in $\mathcal{F}_r$.*

Now we consider the regression problem. Let $n \in \mathbb{N}$, a family of (discrete) functions $\varphi_i \in \mathbb{R}^r$, $1 \leq i \leq n$, and the linear space of functions

$$\mathcal{F}_r := \{\sum_{i=1}^{n} a_i \varphi_i : a_i \in \mathbb{R} \; \forall \; 1 \leq i \leq n\}$$

be given. The minimization of

$$h_r(f_r, z) = \sum_{s \in r}(z_s - \sum_{i=1}^{n} a_i \varphi_i(s))^2$$

in $a \in \mathbb{R}^n$ leads to the normal equations

$$\begin{aligned} \sum_{s \in r} z_s \varphi_j(s) &= \sum_{s \in r} \sum_{i=1}^{n} a_i \varphi_i(s) \varphi_j(s) \,, 1 \leq j \leq n, \\ \iff \sum_{s \in r} z_s \varphi_j(s) &= \sum_{i=1}^{n} a_i \sum_{s \in r} \varphi_i(s) \varphi_j(s) \,, 1 \leq j \leq n. \end{aligned}$$

With the n-element vectors Y and a given by

$$Y := \left(\sum_{s \in r} z_s \varphi_j(s) \right)_{1 \leq j \leq n}$$

and $a := (a_i)_{1 \leq i \leq n}$, and the $n \times n$ matrix M defined by

$$M := \left(\sum_{s \in r} \varphi_i(s) \varphi_j(s) \right)_{1 \leq i,j \leq n},$$

this can be written as the linear equation system

$$Y = Ma. \tag{3.3}$$

Let $\hat{a}$ be a solution of the equation system (3.3). For the computation of the approximation error, the following holds.

$$\min_{f_r \in \mathcal{F}_r} \sum_{s \in r}(z_s - f_r(s))^2 = \sum_{s \in r}(z_s - \sum_{i=1}^{n} \hat{a}_i \varphi_i(s))^2$$

$$= \sum_{s\in r} z_s^2 - 2\sum_{i=1}^{n} \hat{a}_i Y_i + \sum_{i=1}^{n} \hat{a}_i^2 M_{ii}. \qquad (3.4)$$

In the following we denote a minimizer of $\|f_r - z|_r\|_2^2$ by $\Pi_{\mathcal{F}_r} z_r$. Equations (3.3) and (3.4) lead to the following important observation.

Remark 3.1.12: *For efficient computation of a projection $\Pi_{\mathcal{F}_r} z_r$, a fast calculation of the $\frac{n(n+1)}{2} + n$ 'moments' $\sum_s \varphi_i(s)\varphi_j(s)$ and $\sum_{s\in r} z_s\varphi_j(s)$, $1 \le i, j \le n$, is essential! A computation of the approximation error requires an additional computation of the sum $\sum_s z_s^2$. A scheme for a fast computation of these sums for polygonal domain r will be given in Section 3.3.*

Nothing has been said about the uniqueness of the solution of the equation system 3.3. This in general depends on the shape of r. In applications it is sufficient to compute an arbitrary minimizer using a pseudoinverse of M.

Lemma 3.1.13: *Assume that the 'moments' $\sum_s \varphi_i(s)\varphi_j(s)$ and $\sum_{s\in r} z_s\varphi_j(s)$, $1 \le i, j \le n$, are known. Then a minimization of the functional*

$$h_r(f_r, z) = \sum_{s\in r} (z_s - \sum_{i=1}^{n} a_i\varphi_i(s))^2$$

in $a \in \mathbb{R}^n$ can be done with a time complexity of $O(n^3)$ and a spatial complexity of $O(n^2)$.

Proof. The computation of a pseudoinverse M^- of M can be done in $O(n^3)$, compare Trefethen and Bau (1997), pp. 83-85. The solution $a = M^- Y$ takes another n^2 steps. Because for the computation of M^- a fixed number of matrices with maximal size n^2 are needed, the spatial complexity is $O(n^2)$. □

In practice any kind of functions φ_i, $1 \le i \le n$, can be used to generate $\mathcal{F}_r$. Closing this section we give three standard examples of regression.

Example 3.1.14: *Let the function class $\mathcal{F}_r$ be generated by constant functions on r. Then $n = 1$, $\varphi_1 \equiv 1$ and*

$$\mathcal{F}_r := \{f \in \mathbb{R}^r : f(s) = a \;\forall\; s \in r, a \in \mathbb{R}\}.$$

The minimizer is $f \equiv \mu_r := \frac{1}{|r|}\sum_{s\in r} z_i$, *the empirical mean of* z *over* r. *The minimum of* h_r *is thus given by* $h_r(\mu_r, z) = \frac{1}{|r|}\sum_{s\in r}(z_s - \mu_r)^2 =: \sigma_r^2$, *the empirical variance of* z *over* r.

Example 3.1.15: *Now we consider affine regression. With* $n = 3$, $\varphi_1 \equiv 1$, $\varphi_2(s) = s_1$ *and* $\varphi_3(s) = s_2$, $s \in r$, *we get*

$$\mathcal{F}_r := \{f \in \mathbb{R}^r : f(s) = c + as_2 + bs_2 \ \forall s \in r, a, b, c \in \mathbb{R}\}.$$

Minimization of $h_r((a, b, c), z)$ *is equivalent with solving the linear system of equations (all sums are over* $s \in r$*)*

$$\begin{pmatrix} \sum 1 & \sum s_1 & \sum s_2 \\ \sum s_1 & \sum s_1^2 & \sum s_1 s_2 \\ \sum s_2 & \sum s_1 s_2 & \sum s_2^2 \end{pmatrix} \begin{pmatrix} c \\ a \\ b \end{pmatrix} = \begin{pmatrix} \sum z_s s_1 \\ \sum z_s s_2 \\ \sum z_s \end{pmatrix}. \tag{3.5}$$

For a fast computation of the projection $\Pi_{\mathcal{F}_r} z_r$ *and the approximation error,* 10 *'moments' have to be computed.*

Example 3.1.16: *Quadratic regression is obtained with* $n = 6$, $\varphi_1 \equiv 1$, $\varphi_2(s) = s_1$, $\varphi_3(s) = s_2$, $\varphi_4(s) = s_1 s_2$, $\varphi_5(s) = s_1^2$ *and* $\varphi_6(s) = s_2^2$, $s \in r$. *Thus the function space consists of functions of the form* $f(s) = a + bs_1 + cs_2 + ds_1 s_2 + es_1^2 + fs_2^2$, $a, b, c, d, e, f \in \mathbb{R}$. *For a fast computation of the projection* $\Pi_{\mathcal{F}_r} z_r$ *and the approximation error, only* $21 < \frac{n(n+3)}{2} + 1$ *'moments' have to be computed. This is due to identities like* $\varphi_2\varphi_4 = \varphi_5\varphi_3$, *compare the following equation system*

$$\begin{pmatrix} \sum 1 & \sum s_1 & \sum s_2 & \sum s_1 s_2 & \sum s_1^2 & \sum s_2^2 \\ \sum s_1 & \sum s_1^2 & \sum s_1 s_2 & \sum s_1^2 s_2 & \sum s_1^3 & \sum s_1 s_2^2 \\ \sum s_2 & \sum s_1 s_2 & \sum s_2^2 & \sum s_1 s_2^2 & \sum s_1^2 s_2 & \sum s_2^3 \\ \sum s_1 s_2 & \sum s_1^2 s_2 & \sum s_1 s_2^2 & \sum s_1^2 s_2^2 & \sum s_1^3 s_2 & \sum s_1 s_2^3 \\ \sum s_1^2 & \sum s_1^3 & \sum s_1^2 s_2 & \sum s_1^3 s_2 & \sum s_1^4 & \sum s_1^2 s_2^2 \\ \sum s_2^2 & \sum s_1 s_2^2 & \sum s_2^3 & \sum s_1 s_2^3 & \sum s_1^2 s_2^2 & \sum s_2^4 \end{pmatrix} \begin{pmatrix} c \\ a \\ b \\ d \\ e \\ f \end{pmatrix} = \begin{pmatrix} \sum z_s \\ \sum z_s s_1 \\ \sum z_s s_2 \\ \sum z_s s_1 s_z \\ \sum z_s s_1^2 \\ \sum z_s s_2^2 \end{pmatrix}.$$

3.2 Wedge Segmentation

In this section we introduce and characterize the division of rectangles by intersection with a line. This division will be used to split the local minimization on a rectangle into two parts – so called *wedges* – leading to a

nonlinear approximation. If the minimization on each part is a projection on an n-dimensional space, then the nonlinear approximation can be described with $2n + 2$ parameters. Localized phenomena, such as edges or jumps in a two-dimensional signal, may in general be better approximated with a separation of the local minimization into two separate projections than with a significant increase of the dimension for the linear regression. The division of the rectangles will be applied to the fragments of the partitions introduced in Chapter 2. It may be understood as a more generic form of the local approximation, or as a refinement of the partitions. Both points of view have their own benefit. The former is mainly advantageous for the study of algorithms, as in Chapter 2 and in Chapter 4. The latter is better suited for theoretical analysis such as consistency, see Chapter 5.

We start with the definition of wedges, and comment on the interrelation of these divisions on continuous and discrete image domains. Then we introduce digital lines used for a purely discrete specification of wedge divisions on the discrete domain. In Subsection 3.2.3 we introduce the notion of dichotomy, binary functions on the image domain, which we will use to obtain some more characterizations of wedge partitions.

3.2.1 Wedges

In the following let S be either the continuous image domain $S = [0, 1)^2$ or a discrete domain $S = \{s_1, \dots, s_1 + N_1 - 1\} \times \{s_2, \dots, s_2 + N_2 - 1\}$, $s_1, s_2 \in \mathbb{N}$, $N_1, N_2 \in \mathbb{N}_+$.

Consider a rectangle $r \subset S$. In the sequel, let, for each point $x \in \mathbb{R}^2$ and angle $\alpha \in [0, 2\pi)$, the sets $A_{x,\alpha}(r)$ and $B_{x,\alpha}(r)$ be defined by

$$A_{x,\alpha}(r) = \{(u_1, u_2) \in r : (u_2 - x_2)\cos\alpha \geq (u_1 - x_1)\sin\alpha\},$$
$$B_{x,\alpha}(r) = \{(u_1, u_2) \in r : (u_2 - x_2)\cos\alpha < (u_1 - x_1)\sin\alpha\}.$$

The sets $A_{x,\alpha}(r)$ and $B_{x,\alpha}(r)$ are disjoint, connected, and their union is the rectangle r for each $x \in \mathbb{R}^2$ and $\alpha \in [0, 2\pi)$. This leads to the following definition:

Definition 3.2.1: *For all $x \in \mathbb{R}$ and $\alpha \in [0, 2\pi)$ the set*

$$\mathcal{T}_{x,\alpha}(r) := \{A_{x,\alpha}(r), B_{x,\alpha}(r)\} \tag{3.6}$$

will be called a **wedge division** *of r, and elements of $\mathcal{T}_{x,\alpha}(r)$ will be called* **wedges**.

Note that $A_{x,\alpha}(r)$ and $B_{x,\alpha}(r)$ are the points left and right, respectively, of the vector defined by point x and angle α. Compare figure 3.5.

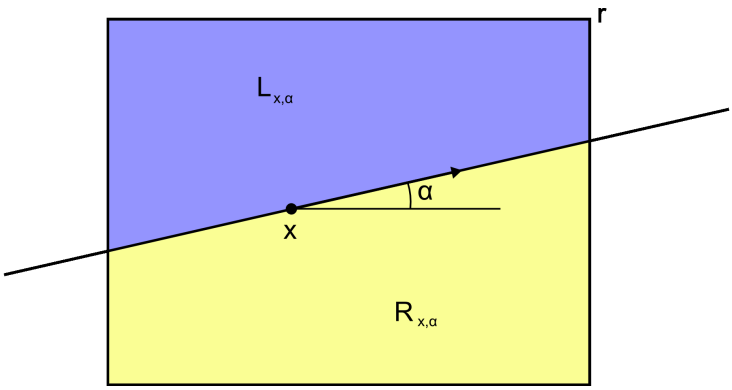


Fig. 3.5: A wedge division of rectangle r.

Remark 3.2.2: *Donoho (1999) introduces wedgelets by splitting rectangles into two pieces by edgels. Edgels are lines connecting the finitely many vertices on the boundary of a given discrete rectangle. We do not make any restrictions now, but will follow up with a different idea of rather global restrictions later on.*

Recall that for a rectangle $R = [s_1, s_1 + n_1) \times [s_2, s_2 + n_2) \cap \mathbb{Z}^2$, $s_1, s_2 \in \mathbb{Z}$, $n_1, n_2 \in \mathbb{N}$ and a set $A \subset [0,1)^2$, the discretization of A in R has been defined by $\boldsymbol{\delta}^R(A) = \{(s_1 - \frac{1}{2} + n_1 \cdot a_1, s_2 - \frac{1}{2} + n_2 \cdot a_2) : (a_1, a_2) \in A\} \cap \mathbb{Z}^2$. As in Sections 2.1 and 2.2, we check if the canonical discretization $\boldsymbol{\delta}^S$ maps wedge divisions to wedge divisions.

Lemma 3.2.3: *Let $S = [s_1, s_1 + N_1) \times [s_2, s_2 + N_2) \cap \mathbb{Z}^2$. For each $x \in \mathbb{R}^2$ and $\alpha \in [0, 2\pi)$ there are $x' \in \mathbb{R}^2$ and $\alpha' \in [0, 2\pi)$ such that for all rectangles $r \subset [0,1)^2$ the following holds:*

$$\boldsymbol{\delta}^S(\mathcal{T}_{x,\alpha}(r)) = \mathcal{T}_{x',\alpha'}(\boldsymbol{\delta}^S(r)).$$

Proof.

A short calculation shows that with the replacements $u_1' = N_1 u_1 + s_1 - \frac{1}{2}$, $u_2' = N_2 u_2 + s_2 - \frac{1}{2}$, $x_1' = s_1 + N_1 x_1 - \frac{1}{2}$ and $x_2' = s_2 + N_2 x_2 - \frac{1}{2}$ the following equation holds:

$$\boldsymbol{\delta}^S(A_{x,\alpha}(r)) = \{(N_1 u_1 + s_1 - \frac{1}{2}, N_2 u_2 + s_2 - \frac{1}{2}) : (u_1, u_2) \in A_{x,\alpha}(r)\}$$

$$= \{(u_1', u_2') \in \boldsymbol{\delta}^S(r) : (u_2' - x_2')\frac{\cos\alpha}{N_2} \geq (u_1' - x_1')\frac{\sin\alpha}{N_1}\}$$

If $\cos\alpha = 0$, then immediately $\boldsymbol{\delta}^S(A_{x,\alpha}(r) = A_{x,\alpha}(\boldsymbol{\delta}^S(r))$. If $\cos\alpha > 0$, then

$$\boldsymbol{\delta}^S(A_{x,\alpha}(r)) = \{(u_1', u_2') \in \boldsymbol{\delta}^S(r) : (u_2' - x_2') \geq (u_1' - x_1')\frac{N_2}{N_1}\tan\alpha\}.$$

The equation holds with '$\leq$', if $\cos\alpha < 0$.

The existence of $\alpha' \in [0, 2\pi)$ with $\tan\alpha = c\tan\alpha'$ for every $c > 0$ and $\operatorname{sgn}(\cos\alpha) = \operatorname{sgn}(\cos\alpha')$ is immediate from $\tan((\frac{\pi}{2}, \frac{3\pi}{2})) = \mathbb{R}$, $\cos((\frac{\pi}{2}, \frac{3\pi}{2})) = [-1, 0)$, $\tan([0, \frac{pi}{2}) \cup (\frac{3\pi}{2}, 2\pi)) = \mathbb{R}$ and $\cos([0, \frac{pi}{2}) \cup (\frac{3\pi}{2}, 2\pi)) = (0, 1]$, see Figure 3.6.

The same arguments work for $B_{x,\alpha}(r)$. Together, this leads to $\boldsymbol{\delta}^S(\mathcal{T}_{x,\alpha}(r)) = \mathcal{T}_{x',\alpha'}(\boldsymbol{\delta}^S(r))$ for appropriate $\alpha' \in [0, 2\pi)$.

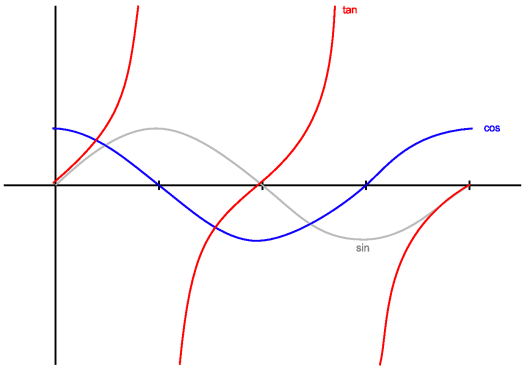


Fig. 3.6: cos and tan

□

In the following we will call a partition that consists of rectangles and wedge divisions of rectangles a **wedge (decorated) partition**. The previous lemma implies that the discretization of a wedge partition is a wedge partition.

Before we comment on the number of wedge divisions of a discrete rectangle, we first consider digital lines for a purely discrete specification of discrete wedges.

3.2.2 Digital Lines

In this subsection we comment on discretization of continuous lines for two reasons. On the one hand, it is a useful preparation for a digital specifica-

tion of wedge divisions. On the other hand, we provide another part of the mathematical background for algorithms doing regression over a polygonal domain. We assume a discrete image domain $S = \{1, \dots, N_1\} \times \{1, \dots, N_2\}$, $N_1, N_2 \in \mathbb{N}$.

For a number $d \in \mathbb{R}$ and an angle $\alpha \in [0, 2\pi)$ consider the following line in $\mathbb{R}^2$:

$$l_{d,\alpha} = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2 : \left\langle \begin{pmatrix} x \\ y \end{pmatrix}, \begin{pmatrix} -\sin\alpha \\ \cos\alpha \end{pmatrix} \right\rangle = d \right\}.$$

Obviously the parameter d is the Euclidean distance of the line to the origin. A natural choice for a discretization of $l_{d,\alpha}$ is a set of points in the grid $\mathbb{Z}^2$ with a given maximal distance δ to the line $l_{d,\alpha}$:

$$L^{\delta}_{d,\alpha} = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{Z}^2 : \; d - \delta \leq \left\langle \begin{pmatrix} x \\ y \end{pmatrix}, \begin{pmatrix} -\sin\alpha \\ \cos\alpha \end{pmatrix} \right\rangle < d + \delta \right\}.$$

Observe that $L^{\delta}_{d,\alpha}$ is the set of all points in $\mathbb{Z}^2$ within a tube around $l_{d,\alpha}$ of width δ where one of the two borders of the tube is included. We will now and then refer to a line of this form as a **digital line**.

The 8-neighborhood of some point $(x_0, y_0) \in \mathbb{Z}^2$ is defined as the set of points $N_8(x_0, y_0) := \{(x, y) \in \mathbb{Z}^2 \setminus (x_0, y_0) : |x - x_0| \leq 1 \text{ and } |y - y_0| \leq 1\}$.

Reasonable requirements on the discretization of lines are:

- For each point $(x, y) \in L$ either $(x', y) \notin L$ for each $x' \neq x$ or $(x, y') \notin L$ for each $y' \neq y$. The lines should thus have a digital thickness of 1 in the horizontal or vertical directions. This is a reasonable assumption for lines that border some area such as a wedge.
- A line should be connected with respect to 8-Neighborhood.

We call a connected set L **minimally connected**, if $L \setminus a$ is not connected for each $a \in L$. The following Lemma contains a choice of $\delta \in \mathbb{R}$ depending on $\alpha \in [0, 2\pi)$ such that $L^{\delta}_{d,\alpha}$ fulfills the aforementioned requirements. Note that the first requirement is fulfilled if an 8-connected line is minimally connected.

Lemma 3.2.4: *Let $d \in \mathbb{R}$ and $\alpha \in [0, 2\pi)$ and $\delta = \max\{|\sin\alpha|/2, |\cos\alpha|/2\}$. Then the line $L^{\delta}_{d,\alpha}$ is minimally connected with respect to 8-neighborhood.*

Proof. Assume, without loss of generality, that $\alpha \in [0, \pi/4]$. The other cases result from reflection on the four axes $y = x$, $y = -x$, $y = 0$ and $x = 0$. Then

$\cos\alpha \geq \sin\alpha \geq 0$ and

$$L^{\delta}_{d,\alpha} = \{(x,y) \in \mathbb{Z}^2 : d - \frac{1}{2}\cos\alpha \leq y\cos\alpha - x\sin\alpha < d + \frac{1}{2}\cos\alpha\}$$
$$= \{(x,y) \in \mathbb{Z}^2 : d' - \frac{1}{2} \leq y - cx < d' + \frac{1}{2}\}$$

with $d' = d/|\cos\alpha|$ and $0 \leq c = \tan\alpha \leq 1$. For each $x \in \mathbb{Z}$ there is one and only one $y \in \mathbb{Z}$ such that $d' + cx - \frac{1}{2} \leq y < d' + cx + \frac{1}{2}$ which is equivalent with $(x,y) \in L^{\delta}_{d,\alpha}$. Therefore we write $y = y(x)$. Now let $(x_0, y(x_0)) \in L^{\delta}_{d,\alpha}$. We show that $|y(x_0) - y(x_0+1)| \leq 1$. From $d' + cx_0 - \frac{1}{2} \leq y(x_0) < d' + cx_0 + \frac{1}{2}$ and $d' + cx + c - \frac{1}{2} \leq y(x_0+1) < d' + cx + c + \frac{1}{2}$ one deduces $|y(x_0) - y(x_0+1)| < c + 1 \leq 2$ which implies $|y(x_0) - y(x_0+1)| \leq 1$. Thus $(x_0+1, y(x_0+1)) \in N_8(x_0, y(x_0))$. Now choose two points $(x_0, y(x_0))$ and $(x_1, y(x_1))$, such that without loss of generality $x_1 - x_0 = n \in \mathbb{N}$. Then the path $(x_0+k, y(x_0+k))_{0\leq k\leq n}$ is a connected path between the two points. Consider $(x_2, y(x_2)) \in L^{\delta}_{d,\alpha}$. Then $L^{\delta}_{d,\alpha} \setminus (x_2, y(x_2))$ is not connected because $|x_1 - x_3| > 1$ for all $x_1 < x_2$ and $x_3 > x_2$. □

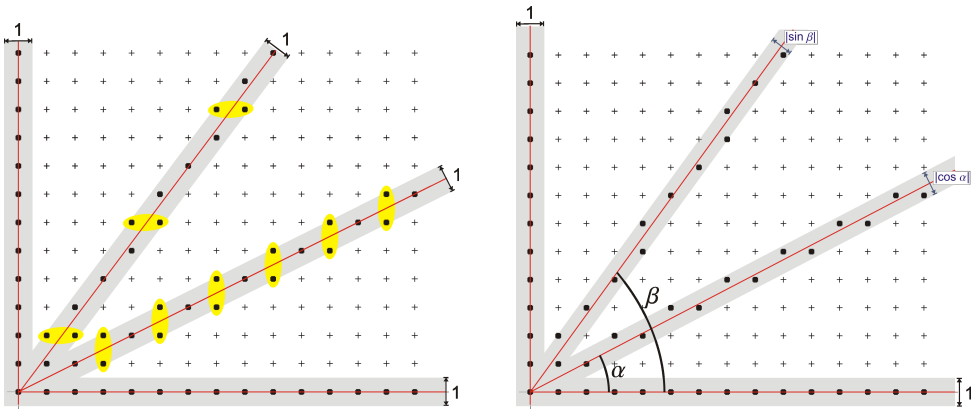


Fig. 3.7: Different choices of δ for the discretization $L^{\delta}_{0,\alpha}$ of lines $l_{0,\alpha}$.

In the following we assume that $\delta = \max\{|\sin\alpha|/2, |\cos\alpha|/2\}$ and write $L_{d,\alpha}$ instead of $L^{\delta}_{d,\alpha}$.

Remark 3.2.5: *The discrete set $L_{d,\alpha}$ is the set of points that are chosen by the famous Bresenham algorithm, a fast algorithm used for drawing lines in computer graphics, see Bresenham (1965) and Rosenfeld and Klette (2001). The somewhat ambiguous choice of tubes for $L^{\delta}_{d,\alpha}$ that are closed on the*

'left' boundary and open on the 'right' corresponds to choosing one of the two possible directions when drawing a line with the Bresenham algorithm. This just specified dependency of the direction is a property that, if unnoted, may cause hard-to-find bugs in computer programs.

Remark 3.2.6: *Assume that* $\alpha \in [0, \pi/4)$. *Then* $\cos\alpha > \sin\alpha > 0$. *Let* $d \in \mathbb{R}$ *and the function* $g : \mathbb{R} \to \mathbb{R}$ *be defined by* $g(x) = \frac{d}{\cos\alpha} + x\frac{\sin\alpha}{\cos\alpha}$. *A short calculation shows that*

$$L_{d,\alpha} = \{(x,y) \in \mathbb{Z}^2 : g(x) - \frac{1}{2} \leq y < g(x) + \frac{1}{2}\} = \{(x, \lceil g(x) - \frac{1}{2} \rceil) : x \in \mathbb{Z}\}.$$

Similar equations apply for the remaining angles $\alpha \in [\pi/4, 2\pi)$*: For 'flat' lines* ($|\sin\alpha| \leq |\cos\alpha|$) *the digital line is thus a set of points in* $\mathbb{Z}^2$ *within a tube with 'vertical diameter'* 1. *For 'steep' lines* ($|\sin\alpha| \geq |\cos\alpha|$) *it is a set of points with 'horizontal diameter' 1.*

Now we establish the link between digital lines and wedges.

Notation 3.2.7: *Let* M *be a subset of* $\mathbb{Z}^2$. *The symbols* $\mathcal{V}^{\triangledown}(M)$ *and* $\mathcal{V}^{\blacktriangle}(M)$ *will in the following denote the following subsets of* $\mathbb{Z}^2$*:*

$$\mathcal{V}^{\blacktriangledown}(M) = \bigcup_{n \in \mathbb{N}_0} \{(x,y) \in \mathbb{Z}^2 : (x, y+n) \in M\},$$
$$\mathcal{V}^{\vartriangle}(M) = \bigcup_{n \in \mathbb{N}} \{(x,y) \in \mathbb{Z}^2 : (x, y-n) \in M, (x,y) \notin M\}.$$

Thus by $\mathcal{V}^{\blacktriangledown}(M)$ we denote the set of points vertically below or in M, and by $\mathcal{V}^{\vartriangle}(M)$ we denote the points strictly above M.

Lemma 3.2.8: *Let* $\delta > 0$, $u \in \mathbb{R}^2$ *and* $\alpha \in [0, \pi/2) \cup (3\pi/2, 2\pi)$ *be given. Let a wedge division* $\mathcal{T}_{u,\alpha}(r) = \{A_{u,\alpha}(r), B_{u,\alpha}(r)\}$ *of a rectangle* r *be given. Then with*

$$d = u_2 \cos\alpha - u_1 \sin\alpha - \delta$$

the equalities $A_{u,\alpha}(r) = r \cap \mathcal{V}^{\blacktriangledown}(L^{\delta}_{d,\alpha})$ *and* $B_{u,\alpha}(r) = r \cap \mathcal{V}^{\vartriangle}(L^{\delta}_{d,\alpha})$ *hold.*

Proof. We prove $A_{u,\alpha}(r) = r \cap \mathcal{V}^{\blacktriangledown}(L^{\delta}_{d,\alpha})$, the second equation can be shown analogously. By assumption $\cos\alpha > 0$ and therefore

$$\mathcal{V}^{\blacktriangledown}(L^{\delta}_{d,\alpha}) = \bigcup_{n \in \mathbb{N}_0} \{(x,y) \in \mathbb{Z}^2 : d - \delta < (y+n)\cos\alpha - x\sin\alpha \leq d + \delta\}$$

$$
\begin{aligned}
&= \{(x,y) \in \mathbb{Z}^2 : \exists n \in \mathbb{N} : d - \delta < (y+n)\cos\alpha - x\sin\alpha \leq d + \delta\} \\
&= \{(x,y) \in \mathbb{Z}^2 : y\cos\alpha - x\sin\alpha \leq u_2\cos\alpha - u_1\sin\alpha\} \\
&= \{(x,y) \in \mathbb{Z}^2 : (y-u_2)\cos\alpha \leq (x-u_1)\sin\alpha\}.
\end{aligned}
$$

Thus $\mathcal{V}^{\blacktriangledown}(L^{\delta}_{d,\alpha}) \cap r = A_{u,\alpha}(r)$. □

Remark 3.2.9: *The previous lemma is very useful for algorithms. It implies that for each wedge division $\mathcal{T}_{u,\alpha}$ of a rectangle there is a* discrete *line $L_{d,\alpha}$ inducing the division. Moreover, if α is chosen such that $\cos\alpha > 0$, the two parts of a wedge division can be determined by taking points on or below a specified digital line and above it. If $\cos\alpha < 0$, then the same is true but the points on the line are then affiliated to the upper part of the division. Because a vertical translation of the digital line by $+1$ makes these points belong to the lower part, the assumption $\cos\alpha \geq 0$ is no restriction with respect to the set of wedge divisions generated by the lines $L^{\delta}_{d,\alpha}$, $d \in \mathbb{R}$, see Figure 3.8.*

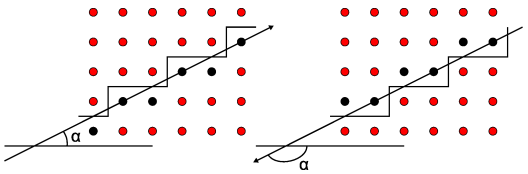


Fig. 3.8: *Wedges generated by the line $l_{d,\alpha}$ and the line $l_{d,\alpha+\pi}$ in opposite direction.*

For algorithms it is useful to have a partition of the discrete set S into disjoint digital lines with the same angle α. This can be achieved with the following definition of lines with angle α and $\delta > 0$:

$$
L^{\delta}_{\delta\cdot d,\alpha} = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{Z}^2 : \ \delta\cdot(d-\frac{1}{2}) \leq \left\langle \begin{pmatrix} x \\ y \end{pmatrix}, \begin{pmatrix} -\sin\alpha \\ \cos\alpha \end{pmatrix} \right\rangle < \delta\cdot(d+\frac{1}{2}) \right\}.
$$

With $\delta(\alpha) := \max\{|\sin\alpha|/2, |\cos\alpha|/2\}$ we define $\bar{L}_{n,\alpha} := L^{\delta(\alpha)}_{\delta(\alpha)n,\alpha}$, $n \in \mathbb{Z}$.

Lemma 3.2.10: *The lines $\bar{L}_{n,\alpha}$, $n \in \mathbb{Z}$, fulfill the following conditions:*

(1) $\bigcup_{n\in\mathbb{Z}} \bar{L}_{n,\alpha} = \mathbb{Z}^2$ *for each* $\alpha \in [0, 2\pi)$,

(2) $\bar{L}_{n,\alpha} \cap \bar{L}_{m,\alpha} = \emptyset$ *for* $n, m \in \mathbb{Z}$ *with* $n \neq m$,

(3) $\bar{L}_{n,\alpha}$ *is minimally connected with respect to* 8*-neighborhood for all* $n \in \mathbb{N}$ *and*

(4) $\bar{L}_{n,\alpha} = \begin{cases} \bar{L}_{0,\alpha} + \binom{n}{0} & \text{if } |\cos\alpha| \geq |\sin\alpha| \\ \bar{L}_{0,\alpha} + \binom{0}{n} & \text{if } |\cos\alpha| \leq |\sin\alpha| \end{cases}$ *for all* $n \in \mathbb{N}$.

Remark 3.2.11: *The conditions in Lemma 3.2.10 have the following implications: For fixed* $\alpha \in [0, 2\pi)$ *and a rectangle* $S \subset \mathbb{Z}^2$*, the set* $\{L_{n,\alpha} \cap S, n \in \mathbb{Z}\}$ *is a disjoint division of* S *into* 8*-connected lines of the same direction. The key feature of this division is that, for fixed* α*, each point in* S *belongs to exactly one digital line. This feature will be crucial for the fast integration algorithm introduced in Section 3.3.*
For some $\alpha \in [0, 2\pi)$ *the set* $\{L_{n,\alpha}, n \in \mathbb{Z}\}$ *is a strict subset of the set of all digital lines with angle* α*. We will consider only the digital lines* $L_{n,\alpha}$*,* $n \in \mathbb{N}$*. This will be the first kind of restriction announced in Remark 3.2.2 that we impose on the set of wedge divisions of rectangles in* S*.*

Proof of Lemma 3.2.10. Conditions (1) and (2) are immediate from the definition of $\bar{L}_{n,\alpha}$. Condition (3) is true by Lemma 3.2.4. We prove condition (4): if $|\cos\alpha| \geq |\sin\alpha|$ then

$$\begin{aligned} \bar{L}_{n,\alpha} &= \begin{cases} \{(x,y) : -\frac{1}{2} \leq (y-n) - x\tan\alpha < \frac{1}{2}\} & \text{if } \cos\alpha > 0, \\ \{(x,y) : -\frac{1}{2} < (y+n) - x\tan\alpha \leq \frac{1}{2}\} & \text{otherwise} \end{cases} \\ &= \begin{cases} \{(x,y+n) : -\frac{1}{2} \leq y - x\tan\alpha < \frac{1}{2}\} & \text{if } \cos\alpha > 0, \\ \{(x,y+n) : -\frac{1}{2} < y - x\tan\alpha \leq \frac{1}{2}\} & \text{otherwise} \end{cases} \\ &= \bar{L}_{0,\alpha} + \binom{0}{n}. \end{aligned}$$

The case $|\cos\alpha| \leq |\sin\alpha|$ can be proved analogously. □

Lemma 3.2.12: *For a pair of points* $s, t \in \mathbb{Z}^2$*,* $s \neq t$*, there is an angle* $\alpha \in [0, 2\pi)$ *and a number* $n \in \mathbb{Z}$ *such that* $s, t \in \bar{L}_{n,\alpha}$*.*

Proof. Assume, without loss of generality, that $0 < t_2 - s_2 < t_1 - s_1$. The other cases result from reflection on the four axes $y = x$, $y = -x$, $y = 0$ and

$x = 0$. Then there is some angle $\alpha \in [0, \pi/4)$ and a value $d \in \mathbb{R}$ such that for the line

$$g : \mathbb{R} \to \mathbb{R}, \quad g(x) = \frac{d}{\cos\alpha} + x\frac{\sin\alpha}{\cos\alpha}$$

the following is true: $g(s_1) = s_2$ and $g(t_1) = t_2$. By Remark 3.2.6 it holds that $L^{\delta}_{d,\alpha} = \{(x, y) \in \mathbb{Z}^2 : g(x) - \frac{1}{2} \leq y < g(x) + \frac{1}{2}\}$. A vertical translation of the line g by $-\frac{1}{2} \leq v < \frac{1}{2}$ results in $s, t \in L_{d+v\cdot\cos\alpha,\alpha}$. Choose $v \in [-\frac{1}{2}, \frac{1}{2})$ such that $\frac{d}{\cos\alpha} + v \in \mathbb{Z}$. □

3.2.3 Dichotomies

As there are only finitely many partitions of a finite set S, there are also only finitely many wedge partitions of S. A continuous line specified by $d \in \mathbb{R}$ and $\alpha \in [0, 2\pi)$ induces one and only one partition. In the previous paragraph we showed that for each wedge partition generated by a line in $\mathbb{R}^2$ there is also one discrete line inducing this partition. In this subsection we will show that to each wedge division of S corresponds exactly one pair of points in S. We will utilize this to give an estimate of the number of wedge divisions of S, which is important for the considerations regarding consistency in Chapter 5.

Koplowitz et al. (1990) have already proved that the number of wedge divisions of a $n \times n$ grid is given by $3n^4/\pi^2 + O(n^3 \log n)$. In Rosenfeld and Klette (2001) a similar result is stated for a $m \times n$ grid, $m \geq n$: the number of partitions can be estimated by $3/\pi^2 m^2 n^2 + O(m^2 n \log n) + O(mn^2 \log\log n)$. They have also developed the correspondence of wedge divisions and pair of points. Their arguments are abstract and do not give a deep insight into details of the discrete geometrical problem. Our constructive derivation will reveal some more structure, and will even yield an algorithm to determine the unique pair of points assigned to the dichotomy.

We start with some notation and a definition.

Notation 3.2.13: *Let two points $a, b \in \mathbb{R}^2$ be given. With $\binom{x}{y}^{\perp} := \binom{-y}{x}$, $x, y \in \mathbb{R}$, we define the following function*

$$\begin{aligned} d_{ab} : S &\to \mathbb{R}, \\ s &\mapsto \left\langle (s-a), (b-a)^{\perp} \right\rangle . \end{aligned}$$

The function d_{ab} measures the signed, weighted distance of points $s \in \mathbb{R}^2$ to the line through a and b. For d_{ab} the following holds: $d_{ab}(s) > 0$ if s is on the 'left' of $\overrightarrow{ab}$, $d_{ab}(s) = 0$ if s is on the line, i.e. there is some $\lambda \in \mathbb{R}$ such that $s = a + \lambda(b - a)$, and $d_{ab}(s) < 0$ if s is on the 'right' of $\overrightarrow{ab}$, see Figure 3.9.

From now on we refer to the case $d_{ab}(s) > 0$ as 'strictly on the left of $\overrightarrow{ab}$' and to $d_{ab}(s) \leq 0$ as 'on the right of $\overrightarrow{ab}$'.

Note that $d_{ab} = -d_{ba}$ and $d_{ab}(s) = d_{sa}(b) = d_{sb}(a)$ and $d_{ab}(a) = d_{ab}(b) = 0$.

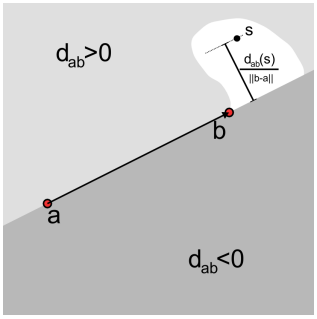


Fig. 3.9: The signed distance function d_{ab} from Notation 3.2.13.

This definition of a dichotomy is a slightly modified version of that of Koplowitz et al. (1990).

Definition 3.2.14 (Dichotomy, Linear Dichotomy, Adjacent Points):
A **Dichotomy** *D on a set S is a mapping that assigns each of the points to one of the two values 0 or 1, $D : S \to \{0, 1\}$. A dichotomy D is called* **linear**, *if there are points a, $b \in \mathbb{R}^2$ such that*

$$D(s) = D_{ab}(s) := \begin{cases} 0 \text{ if } d_{ab}(s) \leq 0, \\ 1 \text{ if } d_{ab}(s) > 0, \end{cases}$$

for all $s \in S$. A dichotomy D is called **nontrivial**, *if $D^{-1}(1) \neq S$ and $D^{-1}(1) \neq \emptyset$. Two points $p, q \in S$, $p \neq q$, are called* **adjacent** *with respect to S, if $\{\lambda p + (1 - \lambda)q, \lambda \in (0, 1)\} \cap S = \emptyset$.*

Obviously, a dichotomy is linear if it can be achieved by a straight line through some $a, b \in \mathbb{R}^2$, and two points are adjacent with respect to S if on the line joining them there is no other point of S. Assume D_{ab} is a linear dichotomy of a rectangle r. Then the set $\{D_{ab}^{-1}(1), D_{ab}^{-1}(0)\}$ is a wedge division of r.

Thus linear dichotomies on some rectangle r correspond to wedge divisions of r.

Before we state the main result about the number of nontrivial linear dichotomies, we need some preparation:

Lemma 3.2.15: *Consider the points $a, b, p, q \in \mathbb{R}^2$ with $d_{ab}(p) \leq 0$ and $d_{ab}(q) > 0$.*

1. *Let $Q \in \mathbb{R}^2$ such that $d_{pq}(Q) > 0$ and $d_{ab}(Q) > 0$. Then the following inclusion holds for all $s \in \mathbb{R}^2$:*

$$d_{ab}(s) > 0 \ \wedge \ d_{pq}(s) \leq 0 \qquad \Longrightarrow \qquad d_{pQ}(s) < 0.$$

2. *Let additionally $r \in \mathbb{R}^2$ such that $d_{ab}(r) \leq 0$ and $d_{pq}(r) > 0$ and $d_{pQ}(r) < 0$, then there is some $R \in \mathbb{R}^2$ with $d_{rQ}(R) = 0$, $d_{pq}(R) = 0$ and $d_{ab}(R) \leq 0$.*

Remark 3.2.16: *More vividly the previous lemma states the following, also observe Figure 3.10: Let two pairs of points a, b and p, q in $\mathbb{R}^2$ be given with q strictly on the left and p on the right of $\overrightarrow{ab}$.*

1. *If a point Q is strictly on the left of both, $\overrightarrow{ab}$ and $\overrightarrow{pq}$, then all points on the right of $\overrightarrow{pq}$ and strictly on the left of $\overrightarrow{ab}$ are strictly on the right of $\overrightarrow{pQ}$.*

2. *If moreover, a point r is given on the right of $\overrightarrow{ab}$, strictly on the left of $\overrightarrow{pq}$ and strictly on the right of $\overrightarrow{pQ}$, then the line $\overrightarrow{rQ}$ intersects the line $\overrightarrow{pq}$ in some point R on the right of $\overrightarrow{ab}$.*

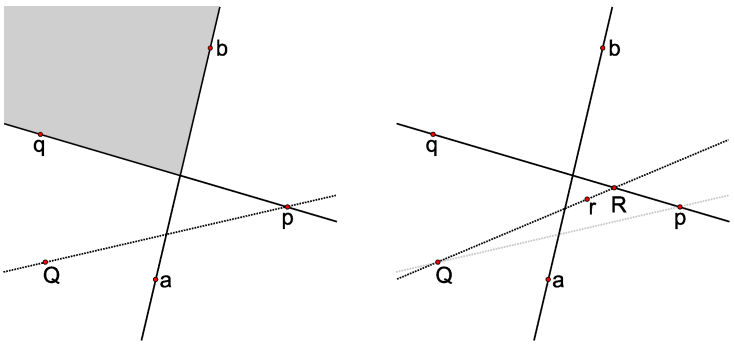


Fig. 3.10: *Illustration for Lemma 3.2.15, case 1 (left) and 2 (right).*

Proof of Lemma 3.2.15. Let $l \in \mathbb{R}^2$ such that $d_{ab}(l) > 0$ and $d_{pq}(l) \leq 0$. We show that this implies $d_{pQ}(l) < 0$. Because $d_{pq}(Q) > 0$ and $d_{pq}(l) \leq 0$ by continuity of L, there is some $0 < \kappa \leq 1$ with $d_{pq}(Q + \kappa(l - Q)) = 0$. Thus there is some $\lambda \in \mathbb{R}$ such that $Q+\kappa(l-Q) = p+\lambda(q-p)$. We show that $\lambda \geq 0$: By $d_{ab}(p) \leq 0$ and $d_{ab}(q) > 0$ there is some $\lambda_0 \geq 0$ with $d_{ab}(p+\lambda_0(q-p)) = 0$, i.e. $d_{ab}(p) = d_{ab}(-\lambda_0(q-p))$. From $0 < d_{ab}(Q+\kappa(l-Q)) = d_{ab}(p+\lambda(q-p)) = d_{ab}((\lambda - \lambda_0)(q-p)) = (\lambda - \lambda_0)d_{ab}(q-p)$ we deduce that $\lambda > \lambda_0 \geq 0$.

Combined there are $\lambda > 0$ and $\kappa \geq 0$ such that $l - Q = \frac{1}{\kappa}(\lambda(q-p)+(p-Q))$, and thus

$$\begin{aligned} d_{pr}(l) &= \frac{1}{\kappa}\left\langle (\lambda(q-p)+(p-Q)), (Q-p)^\perp \right\rangle \\ &= \frac{\lambda}{\kappa}\left\langle q-p, (Q-p)^\perp \right\rangle + \frac{1}{\kappa}\left\langle p-Q, (Q-p)^\perp \right\rangle \\ &= -\frac{\lambda}{\kappa}\left\langle q-p, (Q-p)^\perp \right\rangle = -\frac{\lambda}{\kappa} d_{pq}(Q) < 0, \end{aligned}$$

and the first statement is proved.

Now we prove the second statement: Because $d_{ab}(Q) > 0$, $d_{ab}(p) \leq 0$, $d_{ab}(r) \leq 0$ and $d_{ab}(q) > 0$ there are $0 \leq \lambda_0 < 1$, $0 \leq \mu < 1$ and $0 \leq \nu < 1$ such that $d_{ab}(p+\lambda_0(q-p)) = 0$, $d_{ab}(p+\mu(Q-p)) = 0$ and $d_{ab}(r+\nu(Q-r)) = 0$. Define $r' := r+\nu(Q-r)$. As $p+\lambda_0(q-p)$ and $p+\mu(Q-p)$ and r' are on the line $\overrightarrow{ab}$, there is some $\gamma \in \mathbb{R}$ such that $r' = p+\lambda_0(p-q)+\gamma(\mu(Q-p)-\lambda_0(q-p)))$. First we show that $\gamma < 1$: $d_{pQ}(r') = d_{pQ}(r + \nu(Q-r)) = d_{pQ}((1-\nu)(r)) < 0$ and thus $d_{pQ}(r') = d_{pQ}(p+\lambda_0(p-q)+\gamma(\mu(Q-p)-\lambda_0(q-p))) = d_{pQ}(\lambda_0 q - \gamma\lambda_0 q) = (1-\gamma)\lambda_0 d_{pQ}(q) < 0$ and because $d_{pQ}(q) > 0$ thus $\gamma < 1$. We compute $d_{pq}(r'-Q) = d_{pq}(p-Q+\lambda_0(p-q)+\gamma(\mu(Q-p)-\lambda_0(q-p))) = (\gamma\mu-1)d_{pq}(Q) < 0$. From that and $d_{pq}(r') > 0$ it is immediate that there is some $\vartheta > 0$ such that $d_{pq}(r' + \vartheta(r'-Q)) = 0$. With $d_{ab}(r' + \vartheta(r'-Q)) = -\vartheta d_{ab}(Q) < 0$ the proof is complete. □

Now we are ready to state and prove the central result of this subsection:

Theorem 3.2.17: *The set of nontrivial linear dichotomies of a finite set $S \subset \mathbb{R}^2$ is one to one with the set of adjacent pairs of S.*

Proof. Let p, q be a pair of adjacent points. The mapping $\mathfrak{D}(p,q)$ defined by

$$\mathfrak{D}(p,q)(s) := \begin{cases} 0 \text{ if } d_{pq}(s) < 0 \text{ or } s \in \{p - \lambda(q-p), \lambda \geq 0\}, \\ 1 \text{ if } d_{pq}(s) > 0 \text{ or } s \in \{q + \lambda(q-p), \lambda \geq 0\}, \end{cases}$$

assigns each s in S strictly to the right of $\overrightarrow{pq}$ the value 0, while points strictly to the left of $\overrightarrow{pq}$ are assigned the value 1. If s is on the line, then it gets 1 if it is on the q side and 0 if it is on the p side, see Figure 3.11.

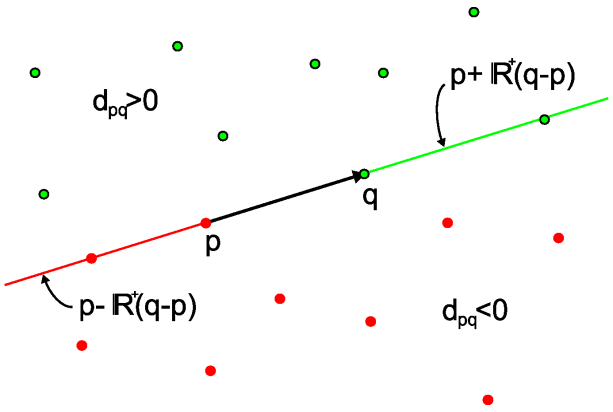


Fig. 3.11: Dichotomy induced by two points p and q.

We show that $\{\mathfrak{D}(p,q) : p,q \in S, p,q \text{ adjacent}\}$ is isomorphic to the set of dichotomies of S, i.e. $\mathfrak{D}(\cdot,\cdot)$ is injective and onto. First we prove that $\mathfrak{D}$ is injective.

Assume that $\mathfrak{D}(p,q) = \mathfrak{D}(r,s)$. Because the relevant expressions are invariant under coordinate transformation, it is sufficient to consider $r = \binom{r_1}{r_2} = \binom{0}{0}$ and $s = \binom{s_1}{s_2} = \binom{a}{0}$, $a > 0$. A short calculation yields

$$\mathfrak{D}(r,s)(t) = \begin{cases} 0 \text{ if } t_2 < 0 \text{ or } (t_2 = 0 \wedge t_1 \leq 0), \\ 1 \text{ if } t_2 > 0 \text{ or } (t_2 = 0 \wedge t_1 \geq a). \end{cases} \tag{3.7}$$

The equations $\mathfrak{D}(r,s)(p) = \mathfrak{D}(p,q)(p) = 0$, $\mathfrak{D}(r,s)(q) = \mathfrak{D}(p,q)(q) = 1$, $\mathfrak{D}(p,q)(r) = \mathfrak{D}(r,s)(r) = 0$ and $\mathfrak{D}(p,q)(s) = \mathfrak{D}(r,s)(s) = 1$ lead to the following conditions:

$$\begin{aligned}
&p_2 > 0 \vee (p_2 = 0 \wedge p_1 \geq a) \implies p_2 \geq 0, \\
&q_2 < 0 \vee (q_2 = 0 \wedge q_1 \leq 0) \implies q_2 \leq 0, \\
&p_1 q_2 - p_2 q_1 > 0 \vee r \in p - \mathbb{R}^+(q-p) \implies p_1 q_2 - p_2 q_1 \geq 0, \\
&p_1 q_2 - p_2 q_1 < a(q_2 - p_2) \vee s \in q + \mathbb{R}^+(q-p) \implies p_1 q_2 - p_2 q_1 \leq a(q_2 - p_2).
\end{aligned}$$

From the last two conditions we get $q_2 \geq p_2$, that together with the first two conditions yields $p_2 = q_2 = 0$ and therefore $p_1 \leq 0$, $q_1 \geq a$. From the second part of the last two conditions we get $p_1 = 0$ and $q_1 = a$. Therefore $p = r$ and $q = s$ and thus the injectivity has been proved.

Now we prove that $\mathfrak{D}$ is onto by induction over a sequence $(S_i)_{i=0,\ldots,|S-2|}$ of increasing subsets of S. Let, as in Definition 3.2.14, a nontrivial linear dichotomy D_{ab} induced by a line through $a \in \mathbb{R}^2$ and $b \in \mathbb{R}^2$ be given. There is a subset $S_0 := \{s_1, s_2\} \subset S$ with $D_{ab}(s_1) = 0$ and $D_{ab}(s_2) = 1$. On S_0 the dichotomy $\mathfrak{D}(s_1, s_2)$ coincides with D_{ab}.

Assume that for a strict subset S_k of S there are points p and q inducing the dichotomy D_{ab} over S_k, i.e. $\mathfrak{D}(p,q)|_{S_k} = D_{ab}|_{S_k}$. Now let another point $Q \in S \setminus S_k$ be given such that, without loss of generality, $D_{ab}(Q) = 0$, then three cases are possible: Either $\mathfrak{D}(p,q)(Q) = 0$, or $\mathfrak{D}(p,q)(Q)$ is undefined, or $\mathfrak{D}(p,q)(Q) = 1$. The cases are treated separately:
If $\mathfrak{D}(p,q)(Q) = 0$, then already $\mathfrak{D}(p,q)|_{S_{k+1}} = D_{ab}|_{S_{k+1}}$, i.e. p and q induce the dichotomy D over $S_{k+1} := S_k \cup \{Q\}$.
If $\mathfrak{D}(p,q)(Q)$ is undefined, then Q is on the line between p and q, i.e. there is some $0 < \lambda < 1$ such that $Q = p + \lambda(p-q)$. Then short calculations show that $d_{pQ} \lesseqgtr 0 \Leftrightarrow d_{pq} \lesseqgtr 0$, $\{q + \lambda(q-p), \lambda \geq 0\} \subset \{Q + \mu(Q-p), \mu \geq 0\}$ and $\{p - \lambda(q-p), \lambda \geq 0\} \subset \{p - \mu(Q-p), \mu \geq 0\}$ and thus $\mathfrak{D}(p,Q)$ has the desired property $\mathfrak{D}(p,Q)|_{S_{k+1}} = D_{ab}|_{S_{k+1}}$.
If $\mathfrak{D}(p,q)(Q) = 1$, then, because $d_{ab}(Q) > 0$ and $d_{pq}(Q) > 0$, we can apply Lemma 3.2.15, and by the finiteness of S_k there is some $P \in S_k$ with $d_{pq}(P) < 0$ such that $\mathfrak{D}(P,Q)|_{S_{k+1}} = D_{ab}|_{S_{k+1}}$. This can be seen as follows: by Lemma 3.2.15 $\{s \in S_k : d_{pq}(s) \leq 0\} \subset \{s \in S_k : d_{P,Q}(s) \leq 0\}$ for all points P such that $d_{ab}(Q) \leq 0$, $d_{pq}(Q) > 0$ and $d_{pQ}(P) \leq 0$. Now applying the first part of lemma 3.2.15 to $P_0 = p$, we can successively choose $P_1, P_2, \ldots, P_n$ until there is no point $s \in S$ such that $d_{ab}(Q) \leq 0$, $d_{pq}(s) > 0$ and $d_{pQ}(P_n) \leq 0$.

□

A consequence of Theorem 3.2.17 is the following raw estimate of the number of linear dichotomies.

Corollary 3.2.18: *The number of linear dichotomies of a grid with n points is bounded from above by $n(n-1)$. The number of digital lines in the grid is bounded from above by $\frac{1}{2}n(n-1)$.*

Proof. Since by Theorem 3.2.17 the number of linear dichotomies of a grid S is given by the number of adjacent pairs $p, q \in S$, the result is immediate. □

Remark 3.2.19: *For a better estimate, the number of* adjacent *pairs needs*

to be estimated for which some number theory is required, compare with Koplowitz et al. (1990) and Acketa and Žunić (1991).

The constructive proof of Theorem 3.2.17 yields an algorithm to determine the unique pair of points (p, q) in S inducing a nontrivial linear dichotomy D_{ab}, see Algorithm 3.1. The algorithm has a complexity of $O(|S|^2)$.

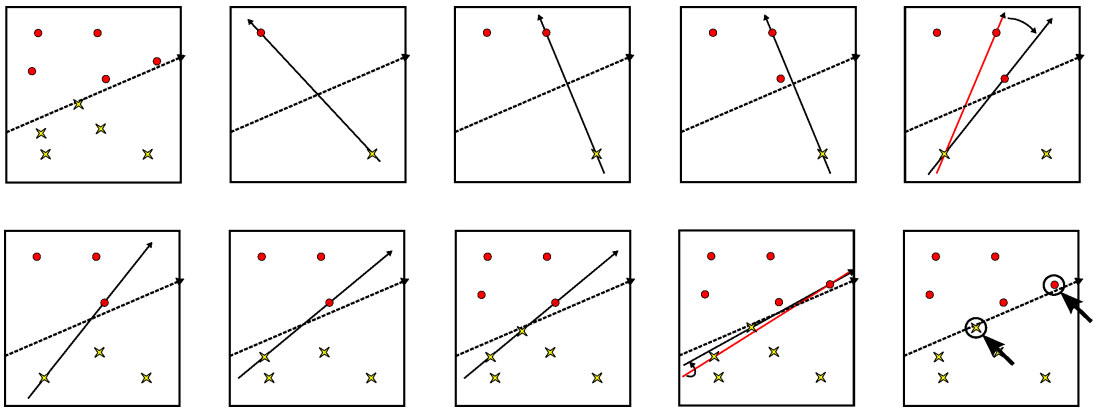


Fig. 3.12: Steps of Algorithm 3.1.

Algorithm 3.1: Determining the Ordered Pair of Points Associated to a Linear Dichotomy

input : set $S = \{s_i, 1 \leq i \leq n\} \subset \mathbb{Z}^2$ with $n \in \mathbb{N}$ elements, linear dichotomy D_{ab} on S
output: pair $p, q \in \mathbb{R}^2$, if $p = inval$ or $q = inval$ then line out of bounds

```
begin
    p ← inval; q ← inval; i ← 1;
    while i ≤ n and (p = inval or q = inval) do
        if D_ab(s_i) = 0 then
            p = s_i
        else
            q = s_i
        end
        i ← i + 1
    end
    if p = inval or q = inval then
        return
    end
    for i = 1 to n do
        d ← D_ab(s_i);
        r^⊥ ← ⟨s_i − a, (p − q)^⊥⟩; r ← ⟨s_i − a, p − q⟩;
        if r^⊥ > 0 or r^⊥ = 0 and r > 0 then z ← 0;
        else if r^⊥ < 0 or r^⊥ = 0 and r ≤ 0 then z ← 1;
        else z ← 2;
        end ;
        if z ≠ d then
            if d = 1 then
                for j = 1 to i do
                    if D_ab(s_j) > 0 then
                        r^⊥ ← ⟨s_i − p, (q − p)^⊥⟩;r ← ⟨s_i − p, (q − p)⟩;
                        if r^⊥ < 0 or r^⊥ = 0 and r > 0 then
                            q ← s_j
                        end
                    end
                end
            else
                for j = 1 to i do
                    if D_ab(s_j) ≤ 0 then
                        r^⊥ ← ⟨s_i − p, (q − p)^⊥⟩;r ← ⟨s_i − p, (q − p)⟩;
                        if r < 0 or r = 0 and ⟨s_i − q, q − p⟩ < 0 then
                            p ← s_j
                        end
                    end
                end
            end
        end
    end
end
```

3.3 Efficient Regression over Polygonal Domains

We continue with the least squares regression problem introduced in paragraph 3.1.3. Consider a finite subset r of the discrete image domain $S \subset \mathbb{Z}^2$, data $z \in \mathbb{R}^r$ and a space $\mathcal{F}_r$ of real valued functions over r defined as the linear hull of functions $\varphi_i : r \to \mathbb{R}$, $1 \leq i \leq n$, $n \in \mathbb{N}$. In Subsection 3.1.3 we showed that an efficient computation of the least squares regression given by the minimization problem $z \xrightarrow{!} \min_{f_r \in \mathcal{F}_r} \sum_{i \in r} (f_r(i) - z_i)^2$ is feasible if the 'moments' $\sum_{s \in r} \varphi_i(s)\varphi_j(s)$ and $\sum_{s \in r} z_s \varphi_j(s)$, $1 \leq i, j \leq n$ can be computed rapidly, compare with Remark 3.1.12.

This section is devoted to the development of an efficient algorithm to integrate any function $f : S \to \mathbb{R}$ over a polygonal domain $r \subset S$. Wedges introduced in Subsection 3.2.1 are polygonal domains. Therefore, in addition to Algorithms 2.3 and 2.8, this algorithm will be another component on the way to efficiently minimize the Potts functional over wedge decorated hierarchical or dyadic partitions.

In Subsection 3.3.1 we start with continuous domain and present some results concerning properties of Jordan curves. Then, utilizing these properties, we comment on efficient integration over the inside of polygons on the continuous domain. After that we carry the results forward to the discrete domain and present an algorithm to efficiently sum up the values of discrete points located in the inside of a polygon.

3.3.1 Jordan Curves

In this paragraph we consider curves in the plane $\mathbb{R}^2$. In the sequel, by the symbol $\mathbb{T}$ we denote the unit circle, $\mathbb{T} := \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 = 1\}$. Recall that a homeomorphism is a bijective and bicontinuous mapping. We need some general results before we can focus on the integration problem.

The following theorem is taken from Dugundi (1974), p.362.

Theorem 3.3.1 (Jordan Curve Theorem): *Let φ be a homeomorphism from $\mathbb{T}$ to $\varphi(\mathbb{T}) \subset \mathbb{R}^2$. Then $\mathbb{R}^2 \setminus \varphi(\mathbb{T})$ has exactly two components, each of which has $\varphi(\mathbb{T})$ as its complete boundary.*

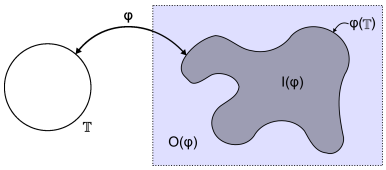


Fig. 3.13: A Jordan Curve

Theorem 3.3.1 leads to the following common definitions.

Definition 3.3.2 (Jordan Curve, Inside and Outside): *Let φ be a homeomorphism from $\mathbb{T}$ to $\varphi(\mathbb{T}) \subset \mathbb{R}^2$. The image $\varphi(\mathbb{T})$ of $\mathbb{T}$ is a* **Jordan Curve** *in $\mathbb{R}^2$. We call the bounded component of $\mathbb{R}^2 \setminus \varphi(\mathbb{T})$ the* **inside** *and the unbounded component the* **outside** *of $\varphi(\mathbb{T})$.*

We will denote the inside of $\varphi(\mathbb{T})$ by $I(\varphi)$ and the outside by $O(\varphi)$. An extension of the Jordan Curve Theorem is the affirmative solution of the so-called Schoenflies Problem, which reads:

Theorem 3.3.3 (Schoenflies Theorem): *Let $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ be a homeomorphism. Then $I(\varphi) \cup \varphi(\mathbb{T})$ is homeomorphic to the two-dimensional disc $\{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\}$.*

Proof. The statement follows immediately from the Riemann mapping theorem of complex analysis. See Dugundi (1974), p. 363 and Rudin (1987), p. 283. □

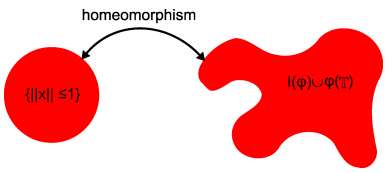


Fig. 3.14: Illustration of the Schoenflies Property

Now we turn to intersections of two Jordan curves. First we give an exact definition of what we mean by 'crossing', 'succeeding crossing' and 'crossing direction'. After that we will present important properties of succeeding crossings of two Jordan curves. These are crucial for an accurate proof of the integration formula.

The circle $\mathbb{T}$ can be identified with the quotient space $\mathbb{R}/\mathbb{Z}$. Thus a homeomorphism $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ can be identified with a 1-periodic function from $\mathbb{R}$ to $\mathbb{R}^2$. By virtue of this identification we will in the sequel use real values as parameters for homeomorphisms.

Lemma 3.3.4: *Consider two homeomorphisms $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ and $\psi : \mathbb{T} \to \psi(\mathbb{T}) \subset \mathbb{R}^2$. Assume that the set*

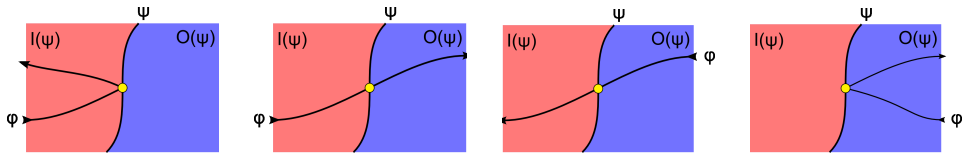
$$K_{\varphi,\psi} = \bigcup_{t\in\mathbb{T}} \{s \in \mathbb{T} : \varphi(s) = \psi(t)\}$$

is finite. Let $M_1 := I(\psi)$ and $M_2 := O(\psi)$, and for all $1 \le i, j \le 2$ let the sets

$$K^{ij}_{\varphi,\psi} = \bigcup_{\varepsilon>0} \{s \in K_{\varphi,\psi} : \varphi((s-\varepsilon, s)) \subset M_i \wedge \varphi((s, s+\varepsilon)) \subset M_j\}$$

be given. Then the sets $K^{ij}_{\varphi,\psi}$, $1 \le i, j \le 2$ are disjoint and $\bigcup_{1\le i,j\le 2} K^{ij}_{\varphi,\psi} = K_{\varphi,\psi}$.

Proof. Because $K_{\varphi,\psi}$ is finite, for all $s \in K_{\varphi,\psi}$ there is some $\varepsilon_0 > 0$ such that $(s-\varepsilon_0, s) \cap K_{\varphi,\psi} = \emptyset$. The set $A := \varphi((s-\varepsilon_0, s))$ does thus not contain a point on $\psi(\mathbb{T})$. By continuity of φ, A is connected and therefore either $A \subset I(\psi)$ or $A \subset O(\psi)$ holds. The same argument holds for an interval of the form $(s, s+\varepsilon)$. □



$s \in K^{11}_{\varphi,\psi}$ $s \in K^{12}_{\varphi,\psi}$ $s \in K^{21}_{\varphi,\psi}$ $s \in K^{22}_{\varphi,\psi}$

Fig. 3.15: Illustration of Lemma 3.3.4: all possible kinds of intersection of two Jordan curves.

The previous Lemma justifies the following

Notation 3.3.5: *With the assumptions of Lemma 3.3.4 we define the function $N_{\varphi,\psi} : \mathbb{T} \to \{-1, 0, 1\}$ by*

$$N_{\varphi,\psi}(s) = \begin{cases} +1 & \text{if } s \in K^{21}_{\varphi,\psi}, \quad \text{('enter')} \\ -1 & \text{if } s \in K^{12}_{\varphi,\psi}, \quad \text{('leave')} \\ 0 & \text{otherwise.} \end{cases}$$

The function $N_{\varphi,\psi}$ admits the following interpretation. At positions $s \in \mathbb{T}$ with $N_{\varphi,\psi}(s) = 0$ the Jordan curve $\varphi(\mathbb{T})$ does not leave the inside or outside of $\psi(\mathbb{T})$. The curves $\varphi(\mathbb{T})$ and $\psi(\mathbb{T})$ may at most be in contact in s. If $N_{\varphi,\psi}(s) = +1$, then the curve $\varphi(\mathbb{T})$ crosses $\psi(\mathbb{T})$ and enters its inside. On points s with $N_{\varphi,\psi}(s) = +1$ the curve $\varphi(\mathbb{T})$ crosses $\psi(\mathbb{T})$ and leaves the inside of $\psi(\mathbb{T})$.

We call points $s \in \mathbb{T}$ with $N_{\varphi,\psi}(s) \neq 0$ **crossings**. The following lemma states that each crossing of $\varphi(\mathbb{T})$ with respect to $\psi(\mathbb{T})$ is a crossing of $\psi(\mathbb{T})$ with respect to $\varphi(\mathbb{T})$.

Lemma 3.3.6: *Consider two homeomorphisms $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ and $\psi : \mathbb{T} \to \psi(\mathbb{T}) \subset \mathbb{R}^2$ with $|\varphi(\mathbb{T}) \cap \psi(\mathbb{T})| < \infty$. Let $s, t \in \mathbb{T}$ be such that $\varphi(s) = \psi(t)$ and $N_{\varphi,\psi}(s) \neq 0$. Then $N_{\psi,\varphi}(t) \neq 0$.*

Proof. We show that the following scenario is impossible: Assume $s, t \in \mathbb{T}$ and $\delta, \varepsilon > 0$ such that $\varphi((s-\varepsilon, s)) \in I(\psi)$, $\varphi((s, s+\varepsilon)) \in O(\psi)$, $\varphi(s) = \psi(t)$, $\psi((t-\delta, t)) \in O(\varphi)$ and $\psi((t, t+\delta)) \in O(\varphi)$.

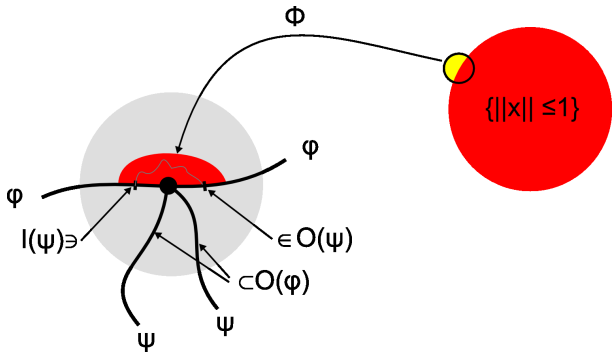


Fig. 3.16: Impossible scenario: $\varphi(\mathbb{T})$ crosses $\psi(\mathbb{T})$ but $\psi(\mathbb{T})$ does not cross $\varphi(\mathbb{T})$.

By continuity of φ and ψ there is some (connected) neighborhood U of $\varphi(s) = \psi(t)$, such that $\psi((s-\varepsilon, s+\varepsilon)) \cup \varphi((t-\delta, t+\delta)) \subset U$. By Theorem 3.3.3 there is a homeomorphism $\Psi : \{x \in \mathbb{R}^2 : \|x\| \leq 1\} \to \varphi(\mathbb{T}) \cup I(\varphi)$. Let $t' := \Psi^{-1}(\psi(t))$, then by continuity of Ψ a ball B around $t \in S^1$ can be chosen such that $\Psi(\{x \in B : \|x\| \leq 1\}) =: U' \subset U$. Moreover, $U' \cap O(\varphi) \neq \emptyset$, $U' \cap I(\varphi) \neq \emptyset$ and $U' \subset O(\varphi)$ and U' is connected, that is U' contains points of $\varphi(\mathbb{T})$, i.e. U contains points of $\psi(\mathbb{T}) \cap I(\varphi)$, which is a contradiction to the assumptions. The case $\varphi((s-\varepsilon, s)) \in O(\psi)$ and $\varphi((s, s+\varepsilon)) \in I(\psi)$ can be treated analogously. □

Before we can state the main theorem of this subsection we need some more preparation:

Lemma 3.3.7: *Consider two homeomorphisms* $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ *and* $\psi : \mathbb{T} \to \psi(\mathbb{T}) \subset \mathbb{R}^2$ *and values* $s_1, s_2, t_1, t_2 \in \mathbb{R}$, $s_1 < s_2$ *and* $t_1 < t_2$ *such that* $\varphi(s_1) = \psi(t_1)$, $\varphi(s_2) = \psi(t_2)$ *and* $\varphi((s_1, s_2)) \cap \psi((t_1, t_2)) = \emptyset$. *Then* $\varphi([s_1, s_2)) \cup \psi((t_1, t_2])$ *is a Jordan curve.*

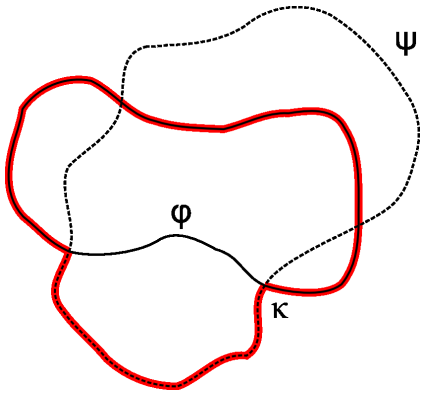


Fig. 3.17: Composition of Jordan curves

Proof. The mapping $\kappa : [0, 1) \to \mathbb{R}^2$ defined by

$$\kappa(z) = \begin{cases} \varphi((1-2z) \cdot s_1 + 2z \cdot s_2) & \text{if } 0 \leq z < \frac{1}{2}, \\ \psi((2-2z) \cdot t_1 + (2z-1) \cdot t_2) & \text{if } \frac{1}{2} \leq z < 1, \end{cases}$$

is bijective and bicontinuous on $[0, 1/2)$ and $[1/2, 1)$ since φ and ψ are homeomorphisms. Because $\varphi(s_1) = \psi(t_1)$ and $\varphi(s_2) = \psi(t_2)$, the left and right limits of κ on $1/2$ and $0 \cong 1$ coincide. As $\kappa((0, 1/2)) \cap \kappa((1/2, 1)) = \emptyset$, the same argument holds for the (existing) inverse of κ. The mapping κ can thus be identified with a homeomorphism from $\mathbb{T}$ to $\mathbb{R}^2$, shortly $\kappa : \mathbb{T} \to \mathbb{R}^2$. Thus $\kappa(\mathbb{T})$ is a Jordan curve. □

Lemma 3.3.8: *Consider two homeomorphisms* $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ *and* $\psi : \mathbb{T} \to \psi(\mathbb{T}) \subset \mathbb{R}^2$ *with* $|\varphi(\mathbb{T}) \cap \psi(\mathbb{T})| < \infty$. *Let* $s \in \mathbb{T}$ *such that* $\varphi(s) \in \psi(\mathbb{T})$ *and* $N_{\varphi,\psi}(s) = 0$. *Then there is a real value* $\varepsilon > 0$ *and a homeomorphism* $\varphi' : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ *such that* $\varphi'(u) = \varphi(u)$ *for all* $u \in \mathbb{T} \setminus (s - \varepsilon, s + \varepsilon)$ *and* $\varphi'((s - \varepsilon, s + \varepsilon)) \cap \psi(\mathbb{T}) = \emptyset$.

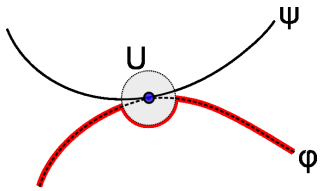


Fig. 3.18: Removal of a non crossing intersection

Proof. Let $t \in \mathbb{T}$ such that $\varphi(s) = \psi(t)$. By Lemma 3.3.6 there is some $\varepsilon > 0$ such that $\varphi((s-\varepsilon, s))$ and $\varphi((s, s+\varepsilon))$ are subsets of the same connected component of ψ, and $\psi((t-\varepsilon, t))$ and $\psi((t, t+\varepsilon))$ are subsets of the same connected component of φ. Moreover, as $\varphi(\mathbb{T})$ and $\psi(\mathbb{T})$ are not self-intersecting, there is some $\delta > 0$ such that for the ball $U = \{z \in \mathbb{R}^2 : \|z - \varphi(t)\| \leq \delta\}$ the following holds: $\varphi(\mathbb{T}) \cap U \subset \varphi((s-\varepsilon, s+\varepsilon))$ and $\psi(\mathbb{T}) \cap U \subset \psi((t-\varepsilon, t+\varepsilon))$. The boundary ∂U of U is a circle and thus a Jordan curve. There are $0 < \varepsilon_1, \varepsilon_2 < \varepsilon$ such that $\varphi(s-\varepsilon_1) \in \partial U$ and $\varphi(s+\varepsilon_2) \in \partial U$. By intersection with $\varphi(\mathbb{T})$ the circle ∂U is divided into two connected parts. The part ∂U_1 of ∂U that is not in the same connected component of $\varphi(\mathbb{T})$ as $\psi((t-\varepsilon, t))$ and $\psi((t, t+\varepsilon))$ does not intersect $\varphi(\mathbb{T})$. Therefore we can apply Lemma 3.3.7, and for a suitable parameterization φ' of the Jordan curve $\partial U_1 \cup \varphi(\mathbb{T} \setminus (s-\varepsilon_1, s+\varepsilon_2))$ it holds that $\varphi'(u) = \varphi(u)$ for all $u \in \mathbb{T} \setminus (s-\varepsilon, s+\varepsilon)$ and $\varphi'((s-\varepsilon, s+\varepsilon)) \cap \psi(\mathbb{T}) = \emptyset$. □

Now we state the main theorem of this subsection:

Theorem 3.3.9: *Consider two homeomorphisms $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ and $\psi : \mathbb{T} \to \psi(\mathbb{T}) \subset \mathbb{R}^2$ with $|\varphi(\mathbb{T}) \cap \psi(\mathbb{T})| < \infty$. Let s_1, s_2, t_1, $t_2 \in \mathbb{R}$ such that $s_1 < s_2$, $\varphi(s_1) = \psi(t_1)$, $\varphi(s_2) = \psi(t_2)$, $N_{\varphi,\psi}(s_1) \neq 0$, $N_{\varphi,\psi}(s_2) \neq 0$ and $N_{\varphi,\psi}(s) = 0$ for all $s_1 < s < s_2$. Then*

(a) $N_{\varphi,\psi}(s_1) = -N_{\varphi,\psi}(s_2) \in \{-1, 1\}$,

(b) $N_{\psi,\varphi}(t_1) = -N_{\psi,\varphi}(t_2) \in \{-1, 1\}$.

Remark 3.3.10: *Theorem 3.3.9 permits the following interpretation: if two Jordan curves defined by the homeomorphisms φ and ψ have finitely many intersections, then 'enter' and 'leave' the inside of ψ toggles on succeeding crossings on $\varphi(\mathbb{T})$ and – less intuitively – the orientation of the curve ψ also toggles on succeeding crossings on $\varphi(\mathbb{T})$, see Figure 3.19.*

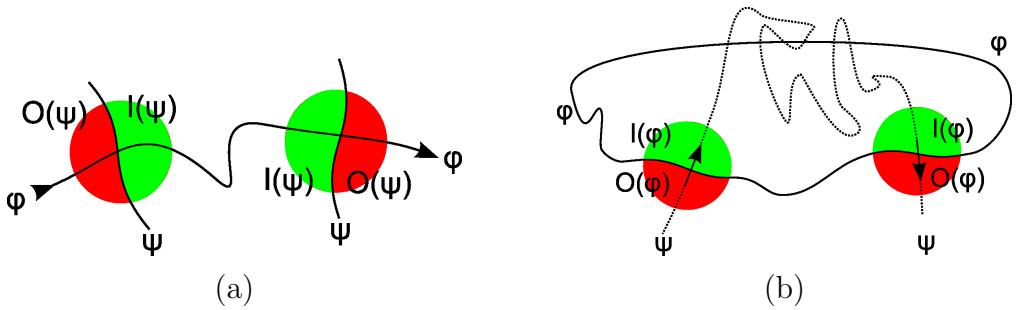


Fig. 3.19: Illustration of Theorem 3.3.9

Proof of Theorem 3.3.9.
(a): Without loss of generality, let $N_{\varphi,\psi}(s_1) = +1$. Then there is some $\varepsilon > 0$ such that $\varphi((s_1, s_1 + \varepsilon)) \subset I(\psi)$. By continuity of φ, the set $\varphi((s_1, s_2))$ is connected and thus $\varphi((s_1, s_2)) \subset I(\psi)$ and therefore $N_\varphi(s_2) = -1$.
(b): Now we show that succeeding crossings in φ have converse directions on ψ. Because on $\varphi((s_1, s_2))$ there are only finitely many (non-crossing) intersections with $\psi(\mathbb{T})$, by Lemma 3.3.8 there is a homeomorphism φ' such that $\varphi'((s_1, s_2)) \cap \psi(\mathbb{T}) = \emptyset$, and φ' coincides with φ up to small neighborhoods of the intersection points $\varphi((s_1, s_2)) \cap \psi(\mathbb{T})$. The mapping $\kappa : [0, 1) \to \mathbb{R}^2$ defined by

$$\kappa(z) = \begin{cases} \varphi'((1-2z) \cdot s_1 + 2z \cdot s_2) & \text{if } 0 \le s < \frac{1}{2}, \\ \psi((2-2z) \cdot t_1 + (2z-1) \cdot t_2) & \text{if } \frac{1}{2} \le s < 1, \end{cases}$$

is a homeomorphism from $\mathbb{T}$ to $\kappa(\mathbb{T}) \subset \mathbb{R}^2$, compare Lemma 3.3.7. There is some $\varepsilon > 0$ such that $\varphi'([s_1 - \varepsilon, s_1 + \varepsilon]) \cap \psi(\mathbb{T}) = \varphi'(s_1)$ and $\varphi'([s_2 - \varepsilon, s_2 + \varepsilon]) \cap \psi(\mathbb{T}) = \varphi'(s_2)$. Now let $a = s_1 - \varepsilon$ and $b = s_2 + \varepsilon$, $a' = s_1 + \varepsilon$ and $b' = s_2 - \varepsilon$.

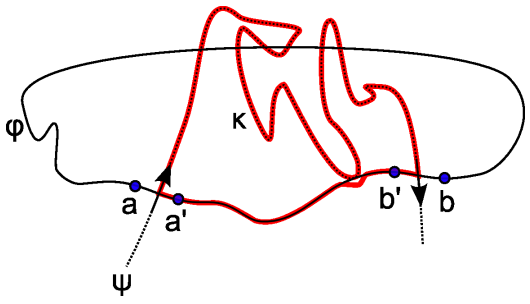


Fig. 3.20: The Jordan curve κ (red, fat)

If a and b are in the same connected component with respect to the Jordan curve $\kappa(\mathbb{T})$, the continuous path between a and b on $\varphi'(\mathbb{T} \setminus [s_1, s_2]) = \varphi(\mathbb{T} \setminus$

$[s_1, s_2]$) has by (a) at most an even number of crossings with $\kappa(\mathbb{T})$. Because φ' is not self intersecting, these crossings must be on $\psi((t_1, t_2))$ and, as there is an even number of such crossings, we have $N_\psi(t_1) = -N_\psi(t_2)$.

In the remainder of the proof we show that, if a and b are in the same connected component with respect to ψ, then a and b are in the same connected component with respect to κ. By assumption $\psi(\mathbb{T})$ does not cross $\kappa(\mathbb{T})$ and thus $\psi(\mathbb{T}) \cap O(\kappa) = \emptyset$ or $\psi(\mathbb{T}) \cap I(\kappa) = \emptyset$. Therefore only the following cases have to be treated.

1. $O(\kappa) \subset I(\psi)$: This case is impossible since $I(\psi)$ is bounded while $O(\kappa)$ is not.

2. $O(\kappa) \subset O(\psi) \implies I(\psi) \subset I(\kappa)$: If $a, b \in I(\psi)$ then $a, b \in I(\kappa)$. If $a, b \in O(\psi)$ then $a', b' \in I(\psi)$ and therefore $a', b' \in I(\kappa)$ which is impossible since $a', b' \in \kappa(\mathbb{T})$.

3. $I(\kappa) \subset I(\psi) \implies O(\psi) \subset O(\kappa)$: If $a, b \in O(\psi)$ then $a, b \in O(\kappa)$. If $a, b \in I(\psi)$ then $a', b' \in O(\psi)$ and therefore $a', b' \in O(\kappa)$ which is impossible since $a', b' \in \kappa(\mathbb{T})$.

4. $I(\kappa) \subset O(\psi) \implies I(\psi) \subset O(\kappa)$: If $a, b \in I(\psi)$ then $a, b \in O(\kappa)$. If $a, b \in O(\psi)$ then $a', b' \in I(\psi)$ and therefore $a', b' \in O(\kappa)$ which is impossible since $a', b' \in \kappa(\mathbb{T})$.

The points a and b are thus in the same component of κ which completes the proof. □

Now we apply the statement of the previous theorem to the case when a vertical line intersects a Jordan curve. Consider a homeomorphism $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$. For points $s \in \mathbb{R}^2$ by $s^{(1)}$ and $s^{(2)}$ we denote the horizontal and vertical component, respectively. Let the horizontal direction function $\mathrm{dir}_\varphi : \mathbb{T} \to \{-1, 0, 1\}$ be defined as

$$\mathrm{dir}_\varphi(s) := \begin{cases} +1 & \text{if } \exists\ \epsilon > 0 : \varphi(s-\delta)^{(1)} < \varphi(s)^{(1)} < \varphi(s+\delta)^{(1)}\ \forall\ 0 < \delta < \epsilon, \\ -1 & \text{if } \exists\ \epsilon > 0 : \varphi(s-\delta)^{(1)} > \varphi(s)^{(1)} > \varphi(s+\delta)^{(1)}\ \forall\ 0 < \delta < \epsilon, \\ 0 & \text{otherwise.} \end{cases}$$

We will use the following results to determine pieces on a vertical line that are in the inside of the Jordan curve.

Corollary 3.3.11: *Let a homeomorphism $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ and a real value $l \in \mathbb{R}$ be given. Assume the set $L = \{s \in \mathbb{T} : \varphi(s)^{(1)} = l\}$ is finite. Let $(s_i)_{1 \le i \le |L|}$ such that $L = \bigcup_{i=1}^{|L|} s_i$ and $s_i^{(2)} > s_{i+1}^{(2)}$ for all $1 \le i < |L|$.*
Then $\operatorname{dir}(s_i) = -\operatorname{dir}(s_{i+1})$ for all $1 \le i < |L|$. Moreover $\{(l, y) : y \in \mathbb{R}\} \cap I(\varphi) = \{\{l\} \times (s_{i+1}^{(2)}, s_i^{(2)}) : \operatorname{dir}(s_i) = \operatorname{dir}(s_1), 1 \le i < |L|\}$.

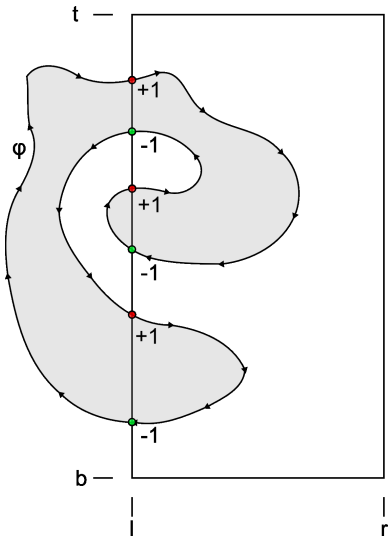


Fig. 3.21: Jordan curve intersecting a vertical line.

Proof. Because $\varphi(\mathbb{T}) \cup I(\varphi)$ is bounded, there are points $b, t, r \in \mathbb{R}$ such that $t > \varphi(s)^{(2)}$, $b < \varphi(s)^{(2)}$ and $r > \varphi(s)^{(1)}$ for all $s \in \mathbb{T}$. The rectangle with vertices (l, b), (l, t), (r, t), (r, b) is a Jordan curve and intersects the curve $\varphi(\mathbb{T})$ only on the line between (l, b) and (l, t). The inside of the rectangle is on the right of this line. Apply Theorem 3.3.9. Because the points vertically above (l, s_1) are in the unbounded component of φ, the second statement is an immediate consequence of the former. □

Now we show that all vertical lines intersecting a Jordan curve have the same directions on the curve indicating whether a vertical line enters or leaves the inside of the Jordan curve.

Corollary 3.3.12: *Let a homeomorphism $\varphi : \mathbb{T} \to \varphi(\mathbb{T}) \subset \mathbb{R}^2$ and real values $l, r \in \mathbb{R}$, $l < r$, be given. Assume the sets $L = \{s \in \mathbb{T} : \varphi(s)^{(1)} = l\}$ and $R = \{s \in \mathbb{T} : \varphi(s)^{(1)} = r\}$ are finite. Let $s_1 \in L$ such that $s_1^{(2)} > s^{(2)}$ for all $s \in L \setminus \{s_1\}$ and $t_1 \in R$ such that $t_1^{(2)} > t^{(2)}$ for all $t \in R \setminus \{t_1\}$. Then $\operatorname{dir}(s_1) = \operatorname{dir}(t_1)$.*

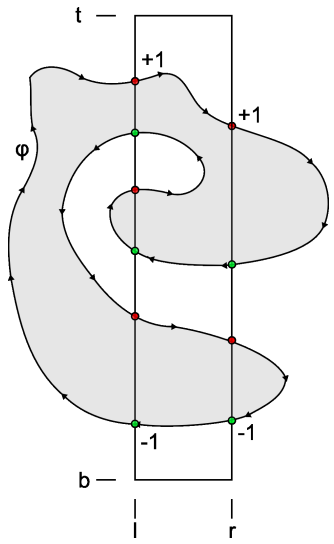


Fig. 3.22: Two vertical lines intersecting a Jordan curve.

Proof. As in the proof of Corollary 3.3.11, we choose $b, t \in \mathbb{R}$ such that $b < \varphi^{(2)}(s) < t$ for all $s \in \mathbb{T}$. The rectangle with vertices (l, b), (l, t), (r, t), (r, b) is a Jordan curve and intersects the curve $\varphi(\mathbb{T})$ only on the lines between (l, b) and (l, t) and between (r, b) and (r, t). The intersection points $\varphi(s_1)$ and $\varphi(t_1)$ are neighbored on the rectangle. Because the inside of the rectangle is on the right of $(l, b) - (l, t)$ and on the left of $(r, b) - (r, t)$, then applying Theorem 3.3.9 yields the result. □

3.3.2 Polygons on the Lattice

In this subsection we apply the results of the previous paragraph to polygons on the lattice. Recall that for a real number x the symbol $\lfloor x \rfloor$ denotes the largest integer less than or equal to x, and $\lceil x \rceil$ is the smallest integer greater than or equal to x.

Definition 3.3.13 (Polygon, Simple Polygon): *Let $n \in \mathbb{N}$ and a family of points $p_i \in \mathbb{R}^2$, $1 \leq i \leq n$, be given. Additionally define $p_{n+1} := p_1$. Consider for all $1 \leq i \leq n$ the functions*

$$\begin{aligned} \varrho_i : [0, 1) &\to \mathbb{R}^2 \\ t &\mapsto (1 - t) \cdot p_i + t \cdot p_{i+1}. \end{aligned}$$

A **polygon** *with vertices* $(p_i)_{1\le i\le n}$ *is the image of the mapping*

$$\begin{aligned}\varrho : [0,1) &\to \mathbb{R}^2\\ s &\mapsto \varrho_{\lceil n\cdot s\rceil}(n\cdot s - \lfloor n\cdot s\rfloor).\end{aligned}$$

A polygon is called **simple**, *if* $\varrho(s)\neq\varrho(t)$ *for all* $s,t\in[0,1)$ *with* $s\neq t$.

In the following we only consider simple polygons, and sometimes denote both the mapping ϱ and its image a polygon.

Remark 3.3.14 (Inner and outside of a Polygon): *From the definition of a simple polygon, it is immediate that its canonical parameterization* $\varrho : [0,1)\to\varrho([0,1))\subset\mathbb{R}^2$ *is bijective and bicontinuous, i.e. simple polygons are Jordan curves. Therefore we can use the Jordan curve definition of the inside and the outside for polygons, and again denote them by* $I(\varrho)$ *and* $O(\varrho)$, *respectively.*

Because a simple polygon is a Jordan curve, the results of the previous paragraph can be immediately applied to simple polygons. For a straightforward use for integration over the inside of the polygons we need some more preparation.

Consider the discrete image domain $S=\{1,\dots,N_1\}\times\{1,\dots,N_2\}$, $N_1,N_2\in\mathbb{N}$ and data $z\in\mathbb{R}^S$. Let $n\in\mathbb{N}$ and a polygon ϱ with vertices p_i and edges ϱ_i, $1\le i\le n$, be given.

Notation 3.3.15: *Similar to Notation 3.2.7 on page 74, we define the subgraphs of the sets* $\varrho_i((0,1))$, $1\le i\le n$, *by*

$$\begin{aligned}\varrho_i^{\triangledown}((0,1)) &= \{s\in S : s^{(1)}=\varrho_i^{(1)}(\lambda), s^{(2)}<\varrho_i^{(2)}(\lambda), \lambda\in(0,1)\},\\ \varrho_i^{\blacktriangledown}((0,1)) &= \{s\in S : s^{(1)}=\varrho_i^{(1)}(\lambda), s^{(2)}\le\varrho_i^{(2)}(\lambda), \lambda\in(0,1)\},\end{aligned}$$

and subgraphs of the points $p\in\mathbb{R}^2$ *by*

$$\begin{aligned}p^{\triangledown} &= \{s\in S : s^{(1)}=p^{(1)}, s^{(2)}<p^{(2)}\},\\ p^{\blacktriangledown} &= \{s\in S : s^{(1)}=p^{(1)}, s^{(2)}\le p^{(2)}\}.\end{aligned}$$

Let $p_0:=p_n$ *and* $p_{n+1}:=p_1$ *and* $z\in\mathbb{R}^S$. *For each* $1\le i\le n$ *we define*

$$\bar{\mathcal{Y}}_i(\varrho,z) := \begin{cases}\displaystyle\sum_{s\in\varrho_i^{\triangledown}((0,1))} z_s & \text{if } p_i^{(1)}<p_{i+1}^{(1)},\\ \displaystyle-\sum_{s\in\varrho_i^{\blacktriangledown}((0,1))} z_s & \text{if } p_i^{(1)}>p_{i+1}^{(1)},\\ 0 & \text{otherwise,}\end{cases}$$

and

$$\mathcal{Y}_i(\varrho, z) := \begin{cases} \sum\limits_{s \in p_i^{\triangledown}} z_s & \text{if } p_{i-1}^{(1)} < p_i^{(1)} < p_{i+1}^{(1)}, \\ -\sum\limits_{s \in p_i^{\blacktriangledown}} z_s & \text{if } p_{i-1}^{(1)} > p_i^{(1)} > p_{i+1}^{(1)}, \\ 0 & \text{otherwise.} \end{cases}$$

We are now ready to state the following

Theorem 3.3.16: *Let $n \in \mathbb{N}$ and a simple polygon ϱ with vertices p_i, $1 \leq i \leq n$, be given. Assume that $p_1^{(1)} < p_2^{(1)}$ and that $p_1^{(2)} \geq p_i^{(2)}$ for all $1 < i \leq n$. Then*

$$\sum_{s \in S \cap I(\varrho)} z_s = \sum_{i=1}^{n} \left(\mathcal{Y}_i(\varrho) + \bar{\mathcal{Y}}_i(\varrho) \right). \tag{3.8}$$

Remark 3.3.17: *Theorem 3.3.16 implies that summing data over the inside of a polygon can be performed efficiently if the vertical sums $\mathcal{Y}_i(\varrho)$ and $\bar{\mathcal{Y}}_i(\varrho)$ can be computed rapidly, or if they are known from some preliminary calculation. Values of edges with a left-right direction are getting added, while edges with opposite direction have to be subtracted. The assumption that $p_1^{(1)} < p_2^{(1)}$ and $p_1^{(2)} \geq p_i^{(2)}$ for all $1 < i \leq n$ is not very restrictive. For each polygon this can be achieved by a suitable permutation of the vertices. The separate treatment of vertices and edges however is crucial. It is for example wrong to consider half closed edges $\varrho_i([0,1))$, $1 \leq i \leq n$, and remove the vertices in the formula in Theorem 3.3.16.*

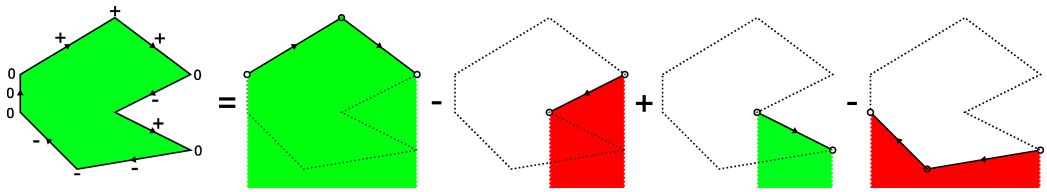


Fig. 3.23: Illustration: $\sum_{s \in S \cap I(\varrho)} z_s = \sum_{i=1}^{n} \mathcal{Y}_i(\varrho) + \bar{\mathcal{Y}}_i(\varrho)$.

Proof of Theorem 3.3.16. The proof has the following structure: We divide S into vertical lines on the integers. Then we apply Corollary 3.3.11 to determine the pieces of each line that is in the inside of ϱ. We show that summing up z on all such pieces on all vertical lines can be done by a subtraction formula, and is equivalent with summing up over lines and points as suggested.

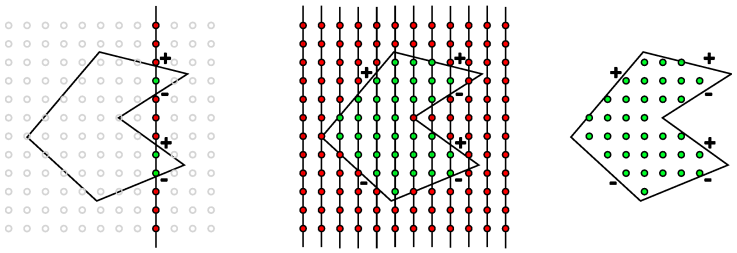


Fig. 3.24: Determining the inside of a polygon with vertical lines

Let for each $x \in \mathbb{Z}$ the crossings of the vertical line at horizontal position x with ϱ be given by $\omega_{x,i}$, $1 \leq i \leq n_x$ such that $\omega_{x,i}^{(2)} > \omega_{x,i+1}^{(2)}$ for all $1 \leq i < n_x$. Consider $\Omega_x := \{\omega_{x,i} : 1 \leq i \leq n_x\}$ and $\Omega := \bigcup_{x\in\mathbb{Z}} \Omega_x$. By assumption and by Corollary 3.3.12 it holds that if $n_x > 0$ then $\mathrm{dir}_\varrho(\omega_{x,1}) = +1$. We apply Corollary 3.3.11 and get

$$\begin{aligned}
\sum_{s\in S\cap I(\varrho)} z_s &= \sum_{x\in\mathbb{Z}} \sum_{i=1}^{n_x/2} \sum_{u\in(\omega_{x,2i+1}^{(2)},\omega_{x,2i}^{(2)})} z_{x,u} = \sum_{x\in\mathbb{Z}} \sum_{i=1}^{n_x/2} \left(\sum_{u<\omega_{x,2i}^{(2)}} z_{x,u} - \sum_{v\leq\omega_{x,2i+1}^{(2)}} z_{x,v} \right) \\
&= \sum_{x\in\mathbb{Z}} \left(\sum_{\substack{\omega\in\Omega_x:\\ \mathrm{dir}_\varrho(\omega)=+1}} \sum_{u<\omega^{(2)}} z_{x,u} - \sum_{\substack{\omega\in\Omega_x:\\ \mathrm{dir}_\varrho(\omega)=-1}} \sum_{v\leq\omega^{(2)}} z_{x,v} \right) \\
&= \sum_{\substack{\omega\in\Omega:\\ \mathrm{dir}_\varrho(\omega)=+1}} \sum_{u<\omega^{(2)}} z_{x,u} - \sum_{\substack{\omega\in\Omega:\\ \mathrm{dir}_\varrho(\omega)=-1}} \sum_{v\leq\omega^{(2)}} z_{x,v}.
\end{aligned}$$

Because for each $\omega \in \Omega$ there is one $1 \leq i \leq n$ such that $\omega \in \varrho_i((0,1))$ or $\omega = p_i$, the following holds:

$$\begin{aligned}
\{\omega \in \Omega : \mathrm{dir}_\varrho(\omega) = +1\} =& \{s \in \varrho_i((0,1)) : s^{(1)} \in \mathbb{Z}, p_i < p_{i+1}, 1 \leq i \leq n\} \\
&\cup \{p_i : p_i^{(1)} \in \mathbb{Z}, p_{i-1} < p_i < p_{i+1}, 1 \leq i \leq n\}.
\end{aligned}$$

An analogous result holds for $\{\omega \in \Omega : \mathrm{dir}_\varrho(\omega) = -1\}$. Therefore

$$\begin{aligned}
\sum_{s\in S\cap I(\varrho)} z_s =& \sum_{\substack{1\leq i\leq n:\\ p_i^{(1)}<p_{i+1}^{(1)}}} \sum_{s\in\varrho_i^{\triangledown}((0,1))} z_s - \sum_{\substack{1\leq i\leq n:\\ p_i^{(1)}>p_{i+1}^{(1)}}} \sum_{s\in\varrho_i^{\blacktriangledown}((0,1))} z_s \\
&+ \sum_{\substack{1\leq i\leq n:\\ p_{i-1}^{(1)}<p_i^{(1)}<p_{i+1}^{(1)}}} \sum_{s\in p_i^{\triangledown}} z_s - \sum_{\substack{1\leq i\leq n:\\ p_{i-1}^{(1)}>p_i^{(1)}>p_{i+1}^{(1)}}} \sum_{s\in p_i^{\blacktriangledown}} z_s \\
=& \sum_{i=1}^{n} \mathcal{Y}_i(\varrho) + \bar{\mathcal{Y}}_i(\varrho).
\end{aligned}$$

The proof is complete. □

3.3.3 Efficient Integration over Polygonal Domain

The integration formula (3.8) in Theorem 3.3.16 can be applied to sum up values on the lattice $\mathbb{Z}^2$ over the inside of polygons consisting of straight lines in the plane $\mathbb{R}^2$. In this subsection we give a very efficient algorithm to integrate values on the lattice over the inside of polygons. Throughout this paragraph we consider the discrete image domain $S := \{1, \ldots, N_1\} \times \{1, \ldots, N_2\}$, $N_1, N_2 \in \mathbb{N}$, and data $z \in \mathbb{R}^S$. Additionally we make frequent use of the symbols $R = (0, N_1 + 1) \times (0, N_2 + 1)$ and $S' := \{0, \ldots, N_1\} \times \{0, \ldots, N_2\}$. Before we state the main theorem of this subsection, a weaker result is given.

Notation 3.3.18: *Recall the definitions $S = \{1, \ldots, N_1\} \times \{1, \ldots, N_2\}$ and $S' = \{0, \ldots, N_1\} \times \{0, \ldots, N_2\}$. Let f be some function from S to $\mathbb{R}$. In the sequel the symbol $I(f)$ will stand for the array $(I_p(f))_{p \in S'}$ given by*

$$I_p(f) := \sum_{s \in p^{\blacktriangledown} \cap S} f(s),\ p \in S'.$$

Lemma 3.3.19: *Let a function $f \in \mathbb{R}^S$ be given. The array $I(f)$ can be computed with a time complexity of $O(N_1 \cdot N_2)$.*

Proof. $I(f)$ can be computed with the recursive formula

$$I_{(x,y)}(f) = \begin{cases} f((x,y)) + I_{(x,y-1)}(f) & \text{if } (x > 0) \text{ and } (y > 0), \\ 0 & \text{otherwise} \end{cases}, \qquad (x,y) \in S'.$$

□

Recall the definition $R = (0, N_1 + 1) \times (0, N_2 + 1)$.

Lemma 3.3.20: *Let a function $f \in \mathbb{R}^S$ be given. Assume that $I(f)$ has already been computed. Then for each simple polygon ϱ with $n \in \mathbb{N}$ vertices $p_i \in R$, $1 \leq i \leq n$, the sum $\sum_{s \in I(\varrho)} f(s)$ can be computed with a time complexity of $O(\sum_{i=1}^{n} |p_i^{(1)} - p_{i+1}^{(1)}|)$ and a space complexity of $O(1)$.*

Proof. Consider the following definitions:

$$\varrho_i^{\blacktriangledown\triangle}((0,1)) := \{s \in S : s^{(1)} = \varrho_i^{(1)}(\lambda), \varrho_i^{(2)}(\lambda) - 1 < s^{(2)} \leq \varrho_i^{(2)}(\lambda), \lambda \in (0,1)\},$$
$$\varrho_i^{\triangledown\blacktriangle}((0,1)) := \{s \in S : s^{(1)} = \varrho_i^{(1)}(\lambda), \varrho_i^{(2)}(\lambda) - 1 \leq s^{(2)} < \varrho_i^{(2)}(\lambda), \lambda \in (0,1)\}.$$

The sets $\varrho_i^{\blacktriangledown\triangle}(0,1)$ $[\varrho_i^{\triangledown\blacktriangle}(0,1)]$ are the sets of points vertically [strictly] below $\varrho_i(0,1)$ with a maximal vertical distance less than [or equal] 1. By the following observation

$$\varrho_i^{\blacktriangledown}((0,1)) = \bigcup_{p \in \varrho_i^{\blacktriangledown\triangle}((0,1))} p^{\blacktriangledown}, \qquad \varrho_i^{\triangledown}((0,1)) = \bigcup_{p \in \varrho_i^{\triangledown\blacktriangle}((0,1))} p^{\blacktriangledown},$$

and as $\varrho_i^{\triangledown\blacktriangle}((0,1)) \subset S'$ and $\varrho_i^{\blacktriangledown\triangle}((0,1)) \subset S'$, the values of $\mathcal{Y}_i(\varrho, f)$, $1 \leq i \leq n$, defined in Notation 3.3.15 and used in Theorem 3.3.16 can be computed with the formula

$$\bar{\mathcal{Y}}_i(\varrho, f) = \begin{cases} \sum_{p \in \varrho_i^{\triangledown\blacktriangle}((0,1))} I_p(f) & \text{if } p_i^{(1)} < p_{i+1}^{(1)}, \\ -\sum_{p \in \varrho_i^{\blacktriangledown\triangle}((0,1))} I_p(f) & \text{if } p_i^{(1)} > p_{i+1}^{(1)}, \\ 0 & \text{otherwise .} \end{cases}$$

Because $|\varrho_i^{\triangledown\blacktriangle}| \leq |p_i^{(1)} - p_{i+1}^{(1)}|$ and $|\varrho_i^{\blacktriangledown\triangle}| \leq |p_i^{(1)} - p_{i+1}^{(1)}|$, Theorem 3.3.16 together with this formula provides a scheme for summing up values in the inside of a polygon ϱ with a time complexity of $O(\sum_{i=1}^{n} |p_i^{(1)} - p_{i+1}^{(1)}|)$ □

An analogous scheme can also be performed to compute the sum of points in the inside *including the boundary* of ϱ, see Figure 3.25.

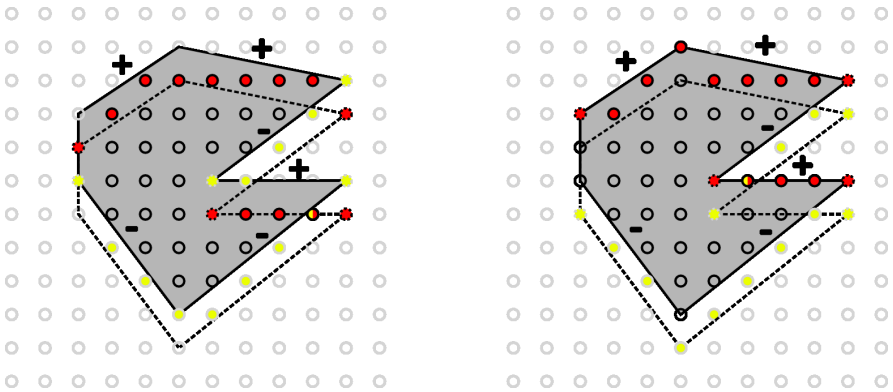


Fig. 3.25: Inside (left) and inside including boundary (right) of a continuous polygon using digital lines 'below' the polygon.

A summary of the two previous Lemmas reads:

Corollary 3.3.21: *Let a function $f \in \mathbb{R}^S$ be given. After a computational preprocessing with a time- and space-complexity of $O(N_1 \cdot N_2)$ the following is possible: For each simple polygon ϱ with $n \in \mathbb{N}$ vertices $p_i \in R$, $1 \leq i \leq n$, the sum $\sum_{s \in I(\varrho)} f(s)$ can be computed with a time complexity of $O(\sum_{i=1}^{n} |p_i^{(1)} - p_{i+1}^{(1)}|)$ and a space complexity of $O(1)$.*

Notation 3.3.22: *For $\alpha \in (-\pi, \pi)$ and $k \in \mathbb{Z}$, the symbol $g_{k,\alpha}$ will denote the mapping*

$$\begin{aligned} g_{k,\alpha} : \mathbb{R} &\to \mathbb{R}, \\ x &\mapsto (k + \frac{1}{2}) \max\{1, |\tan\alpha|\} + x \tan\alpha. \end{aligned}$$

Additionally, for each set $M \subset \mathbb{R}$ we define the subgraphs of $g_{k,\alpha}(M)$ by

$$\begin{aligned} g_{k,\alpha}^{\blacktriangledown}(M) &= \{(x,y) \in S : y \leq g_{k,\alpha}(x), x \in M\}, \\ g_{k,\alpha}^{\triangledown}(M) &= \{(x,y) \in S : y < g_{k,\alpha}(x), x \in M\}. \end{aligned}$$

For each $\alpha \in (-\pi/2, \pi/2]$ and $k \in \mathbb{Z}$ the graph $G_{k,\alpha} \subset \mathbb{R}^2$ is defined as

$$G_{k,\alpha} = \begin{cases} \{(k,y) : y \in \mathbb{R}\} & \text{if } \alpha = \pi, \\ \{(x, g_{k,\alpha}(x)) : x \in \mathbb{R}\} & \text{if } \alpha \neq \pi. \end{cases}$$

For a finite set of angles $\Delta \subset (-\pi/2, \pi/2]$, $n \in \mathbb{N}$ and $R \subset \mathbb{R}^2$, let in the following $\Lambda_n(\Delta, R)$ denote the set of simple polygons with n vertices $p_1, \ldots, p_n$ and $p_{n+1} := p_1$ for which for all $1 \leq i \leq n$ there is some $\alpha \in \Delta$ and $k \in \mathbb{Z}$ such that $p_i, p_{i+1} \in G_{k,\alpha}$. Additionally let $\Lambda(\Delta, R) := \bigcup_{k \in \mathbb{N}} \Lambda_k(\Delta, R)$.

The set $\Lambda_n(\Delta, R)$ consists of all simple polygons with vertices in R that arise from an intersection of lines with angles from the set Δ, where each line has an offset of one half on the y-axis plus some integer on the y- or the x-axis, if it is flat ($|\cos\alpha| \geq |\sin\alpha|$) or steep ($|\sin\alpha| \geq |\cos\alpha|$), respectively. An example is displayed in Figure 3.26.

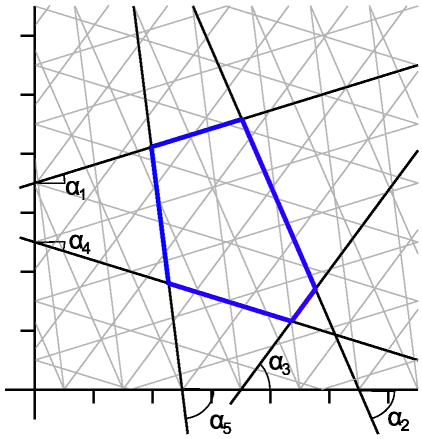


Fig. 3.26: A polygon $\varrho \in \Lambda_5(\{\alpha_1, \ldots, \alpha_5\}, R)$.

Before the necessary components for the main theorem are displayed, we motivate the idea of the fast integration scheme with the following:

Remark 3.3.23: *Let a function $f \in \mathbb{R}^S$, an angle $\alpha \in (-\pi/2, \pi/2)$ and a number $k \in \mathbb{Z}$ be given . Recall the array $I(f) = (I_p(f))_{p \in S'}$ containing the vertical sums of values of f. Consider the array $r = (r_x)_{0 \le x \le N_1}$ with*

$$r_x = \sum_{s \in S \cap g^{\blacktriangledown}_{k,\alpha}([0,x])} f(s),\ 0 \le x \le N_1.$$

Recall that

$$\lfloor x \rfloor = \max\{y \in \mathbb{Z} : y \le x\}, \qquad \lfloor x \rfloor + 1 = \min\{y \in \mathbb{Z} : y > x\},$$
$$\lceil x \rceil = \min\{y \in \mathbb{Z} : y \ge x\}, \qquad \lceil x \rceil - 1 = \max\{y \in \mathbb{Z} : y < x\}.$$

The array $(r_x)_{0 \le x \le N_1}$ can be computed recursively for each $1 \le x \le N_1$ by

$$r_x = \begin{cases} 0 & \text{if } x = 0, \\ r_{x-1} + I_{(x, \lfloor g_{k,\alpha}(x) \rfloor]}(f) & \text{if } g_{k,\alpha}(x) \in [0, N_2 + 1), \\ r_{x-1} + I_{(x,N_2)}(f) & \text{if } g_{k,\alpha}(x) \ge N_2 + 1, \\ r_{x-1} & \text{otherwise.} \end{cases}$$

Now assume real values $a, b \in (0, N_1 + 1)$ with $a < b$. Then the sum of values of the array z in the subgraph $g^{\blacktriangledown}_{k,\alpha}((a,b))$ can be computed with the following formula

$$\sum_{s \in g^{\blacktriangledown}_{k,\alpha}((a,b))} f(s) = \sum_{s \in g^{\blacktriangledown}_{k,\alpha}([0,b))} f(s) - \sum_{s \in g^{\blacktriangledown}_{k,\alpha}([0,a])} f(s).$$

Because $g^{\blacktriangledown}_{k,\alpha}([0,b)) = g^{\blacktriangledown}_{k,\alpha}([0,\lceil b\rceil - 1))$ *and* $g^{\blacktriangledown}_{k,\alpha}([0,a]) = g^{\blacktriangledown}_{k,\alpha}([0,\lfloor a\rfloor))$ *,and as* $\lceil b\rceil - 1 \in [0,N_1]$ *and* $\lfloor a\rfloor \in [0,N_1]$*, this can be written as*

$$\sum_{s\in g^{\blacktriangledown}_{k,\alpha}((a,b))} f(s) = r_{\lceil b\rceil -1} - r_{\lfloor a\rfloor}.$$

For a single line the sum $\sum_{s\in g^{\blacktriangledown}_{pq}([0,x])} f(s)$ *can thus be computed in* $O(1)$ *if the corresponding array* r *has been computed beforehand. See also Figure 3.27.*

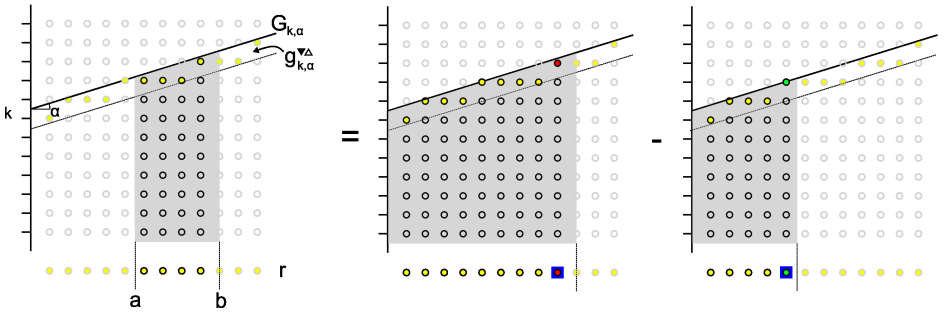


Fig. 3.27: *Illustration of the summation trick* $\sum_{s\in g^{\blacktriangledown}_{k,\alpha}((a,b))} f(s) = r_{\lceil b\rceil -1} - r_{\lfloor a\rfloor}$*. Yellow filled points correspond to the storage array* r*. The red filled point corresponds to* $\lceil b\rceil - 1$*, the green one to* $\lfloor a\rfloor$*.*

Now we define the cumulative sum arrays for arbitrary angles $\alpha \in (-\pi/2, \pi/2)$:

Notation 3.3.24: *Let* $\alpha \in (-\pi/2,\pi/2)$ *and* $f \in \mathbb{R}^S$*. For each* $k \in \mathbb{Z}$ *consider the numbers*

$$l_{k,\alpha} = \min\{x \in \{1,\dots,N_1\} : g_{k,\alpha}(x) \in (0, N_2+1)\},$$
$$r_{k,\alpha} = \max\{x \in \{1,\dots,N_1\} : g_{k,\alpha}(x) \in (0, N_2+1)\}.$$

In the sequel the arrays $I^{(\alpha)}(f) = (I^{(\alpha)}_s(f))_{s\in S}$ *and* $J^{(\alpha)}(f) = (J^{(\alpha)}_s(f))_{s\in S}$ *are defined as follows: For each* $k \in \mathbb{Z}$ *and* $x \in \mathbb{Z}$ *with* $l_{k,\alpha} \le x \le r_{k,\alpha}$ *let*

$$I^{(\alpha)}_{(x,\lfloor g_{k,\alpha}(x)\rfloor)}(f) = \sum_{s\in g^{\blacktriangledown}_{k,\alpha}([l_{k,\alpha},x])} f(s), \qquad J^{(\alpha)}_{(x,\lfloor g_{k,\alpha}(x)\rfloor)}(f) = \sum_{s\in g^{\triangledown}_{k,\alpha}([l_{k,\alpha},x])} f(s).$$

Additionally let $I^{(\pi/2)}(f) = J^{(\pi/2)}(f) = I(f)$*.*

Lemma 3.3.25: *Let* $\alpha \in (-\pi/2,\pi/2)$ *and* $f \in \mathbb{R}^S$*. The arrays* $I^{(\alpha)}(f)$ *and* $J^{(\alpha)}(f)$ *can be computed with a time complexity of* $O(|S|)$*.*

Proof. The following recursive formulas apply for $x \in \mathbb{Z}$ with $l_{k,\alpha} \leq x \leq r_{k,\alpha}$. We omit the parameter f.

$$I^{(\alpha)}_{(x,\lfloor g_{k,\alpha}(x)\rfloor)} = \begin{cases} I_{(x,\lfloor g_{k,\alpha}(x)\rfloor)} & \text{if } x = l_{k,\alpha}, \\ I_{(x,\lfloor g_{k,\alpha}(x)\rfloor)} + I^{(\alpha)}_{(x-1,\lfloor g_{k,\alpha}(x-1)\rfloor)} & \text{otherwise,} \end{cases}$$

$$J^{(\alpha)}_{(x,\lfloor g_{k,\alpha}(x)\rfloor)} = \begin{cases} I_{(x,\lceil g_{k,\alpha}(x)\rceil-1)} & \text{if } x = l_{k,\alpha}, \\ I_{(x,\lceil g_{k,\alpha}(x)\rceil-1)} + J^{(\alpha)}_{(x-1,\lfloor g_{k,\alpha}(x-1)\rfloor)} & \text{otherwise.} \end{cases}$$

Therefore traversing S in an appropriate way yields $|S|$ operations to compute $I^{(\alpha)}$ and $J^{(\alpha)}$. □

Lemma 3.3.26: *Let $\Delta \subset (-\pi/2, \pi/2]$ be finite, $R = (0, N_1+1) \times (0, N_2+1)$ and $f \in \mathbb{R}^S$. Assume that for all $\alpha \in \Delta$ the cumulative sum arrays $I^\alpha(f)$ and $J^{(\alpha)}(f)$ have already been computed. Then for all $n \in \mathbb{N}$ and each polygon $\varrho \in \Lambda_n(\Delta, R)$ the sum $\sum_{s\in I(\varrho)} z_s$ can be computed with a time complexity of $O(n)$ and a space complexity of $O(1)$.*

Proof. We continue with the idea displayed in Remark 3.3.23. For each $k \in \mathbb{Z}$ the line $G_{k,\alpha}$ is an upper boundary of the line $\bar{L}_{k,\alpha}$ defined in Subsection 3.2.2. With $\delta' = \max\{1, |\tan\alpha|\}$ the formula

$$\begin{aligned} \bar{L}_{k,\alpha} &= \{(x,y) \in \mathbb{Z}^2 : (k-\frac{1}{2})\delta' < y - x\tan\alpha \leq (k+\frac{1}{2})\delta'\} \\ &= \{(x,y) \in \mathbb{Z}^2 : g_{k,\alpha}(x) - \delta' < y \leq g_{k,\alpha}(x)\} \\ &\supseteq \{(x,y) \in \mathbb{Z}^2 : y = \lfloor g_{k,\alpha}(x)\rfloor\} \\ &= g^{\blacktriangledown\triangle}_{k,\alpha}(\mathbb{R}) \end{aligned}$$

implies by Lemma 3.2.10 that the sets $g^{\blacktriangledown\triangle}_{g,\alpha}(\mathbb{R})$, $k \in \mathbb{Z}$, are pairwise disjoint. Therefore for each line number k the digital sets $g^{\blacktriangledown\triangle}_{k,\alpha}(\mathbb{R}) \cap R$ are a good place to store the information needed for fast computation of sums of f within subgraphs of the corresponding line $g_{k,\alpha}$. We thus use the same summation trick as in Remark 3.3.23, but use the corresponding values stored in the arrays $I^{(\alpha)}(f) = (I^{(\alpha)}_s(f))_{s\in S'}$ and $J^{(\alpha)}(f) = (J^{(\alpha)}_s(f))_{s\in S'}$.

Recall the definition

$$\begin{aligned} l_{k,\alpha} &= \min\{x \in \{1,\dots,N_1\} : g_{k,\alpha}(x) \in (0, N_2+1)\}, \\ r_{k,\alpha} &= \max\{x \in \{1,\dots,N_1\} : g_{k,\alpha}(x) \in (0, N_2+1)\}. \end{aligned}$$

Let $(x_1, y_1), (x_2, y_2) \in G_{k,\alpha} \cap R$. Assume that $x_1 < x_2$. Then $x_1, x_2 \in (l_{k,\alpha} - 1, r_{k,\alpha} + 1)$ and the following holds: Let $X_1 = \lfloor x_1 \rfloor$, $Y_1 = \lfloor g_{k,\alpha}(X_1) \rfloor$, $X_2 = \lceil x_2 - 1 \rceil$ and $Y_2 = \lfloor g_{k,\alpha}(X_2) \rfloor$. If $X_2 \leq x_1$ then $X_2 = X_1$ and there are no integers between x_1 and x_2. But if $X_2 > x_1$ then $l_{k,\alpha} \leq X_2 \leq r_{k,\alpha}$ and therefore $0 \leq Y_2 \leq N_2$. If $l_{k,\alpha} \leq X_1 \leq r_{k,\alpha}$ then $0 \leq Y_1 \leq N_2$ and thus $X_1 < l_{k,\alpha}$ if $Y_1 \notin [0, N_2]$. Let $\mathcal{Z}_\varrho(a,b) := \sum_{s \in g^{\triangledown}_{k,\alpha}((a,b))} f(s)$ and $\mathcal{Z}'_\varrho(a,b) := \sum_{s \in g^{\blacktriangledown}_{k,\alpha}((a,b))} f(s)$. Then

$$\mathcal{Z}_\varrho(a,b) = \begin{cases} 0 & \text{if } X_2 = X_1, \\ I^{(\alpha)}_{(X_2,X_2)}(f) - I^{(\alpha)}_{(X_1,Y_1)}(f) & \text{if } X_2 \neq X_1 \text{ and } 0 \leq Y_1 \leq N_2, \\ I^{(\alpha)}_{(X_2,Y_2)}(f) & \text{otherwise,} \end{cases}$$

$$\mathcal{Z}'_\varrho(a,b) = \begin{cases} 0 & \text{if } X_2 = X_1, \\ J^{(\alpha)}_{(X_2,X_2)}(f) - J^{(\alpha)}_{(X_1,Y_1)}(f) & \text{if } X_2 \neq X_1 \text{ and } 0 \leq Y_1 \leq N_2, \\ J^{(\alpha)}_{(X_2,Y_2)}(f) & \text{otherwise.} \end{cases}$$

Let now a polygon $\varrho \in \Lambda_n(\Delta, R)$ be given. Assume that the corresponding angles $\alpha_i \in [-\pi/2, \pi/2)$, the line offsets $k_i \in \mathbb{Z}$, $1 \leq i \leq n$ and vertices p_i, $1 \leq i \leq n$, $p_{n+1} := p_1$, $p_0 := p_n$, are given such that $p_i \in g_{k_i,\alpha_i}$ and $p_{i+1} \in g_{k_i,\alpha_i}$. Let $x_i := p_i^{(1)}$ for all $1 \leq i \leq n$. Then with

$$\bar{\mathcal{Y}}_i(\varrho) = \begin{cases} \mathcal{Z}_\varrho(x_i, x_{i+1}) & \text{if } x_i < x_{i+1}, \\ -\mathcal{Z}'_\varrho(x_{i+1}, x_i) & \text{if } x_i > x_{i+1}, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\mathcal{Y}_i(\varrho) = \begin{cases} I_{x_i, \lfloor g_{k_i,\alpha_i}(x_i) \rfloor} & \text{if } x_i \in \mathbb{Z} \text{ and } x_{i-1} < x_i < x_{i+1}, \\ I_{x_i, \lceil g_{k_i,\alpha_i}(x_i) \rceil - 1} & \text{if } x_i \in \mathbb{Z} \text{ and } x_{i-1} > x_i > x_{i+1}, \\ 0 & \text{otherwise,} \end{cases}$$

by Theorem 3.3.16 the sum of values of z in the inside of ϱ is given by

$$\sum_{s \in S \cap I(\varrho)} z_s = \sum_{i=1}^{n} \left(\mathcal{Y}_i(\varrho) + \bar{\mathcal{Y}}_i(\varrho) \right). \tag{3.9}$$

Because the $|\Delta|$ matrices I^α and J^α, $\alpha \in \Delta$, can be computed in $O(N_1 \cdot N_2)$, and because the computation of $\mathcal{Y}_i$ and $\bar{\mathcal{Y}}_i$ can be done in $O(1)$ for each $1 \leq i \leq n$ the lemma is proved. □

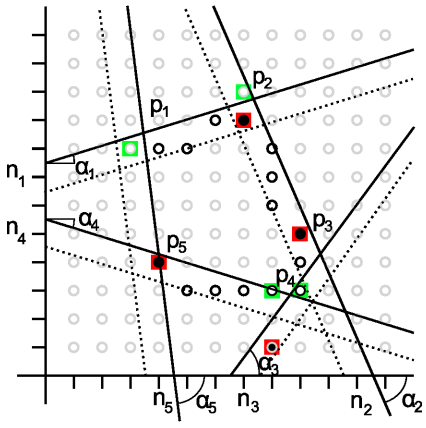


Fig. 3.28: Illustration of the summation formula (3.9), black filled points (red underneath) correspond to the '+' part of $\mathcal{Y}_i(\varrho)$ and $\bar{\mathcal{Y}}_i(\varrho)$ and white points (green underneath) correspond to '-'.

The following Theorem is an immediate consequence of the previous two lemmas and is the main result of this Section:

Theorem 3.3.27: *Let $\Delta \subset (-\pi/2, \pi/2]$ be finite, $f \in \mathbb{R}^S$ and $R = (0, N_1 + 1) \times (0, N_2 + 1)$. After a computational preprocessing with time- and space-complexity of $O(N_1 \cdot N_2 \cdot |\Delta|)$ the following is possible:*
For all $n \in \mathbb{N}$ and each polygon $\varrho \in \Lambda_n(\Delta, R)$ the sum $\sum_{s \in I(\varrho)} f(s)$ can be computed with a time complexity of $O(n)$ and a spatial complexity of $O(1)$.

The construction used in Remark 3.3.23 and the proof of Lemma 3.3.26 leads to an algorithm to determine the sum of values of some data $z \in \mathbb{R}^S$ in the inside of some polygon $\varrho \in \Lambda_n(\Delta, R)$. For simplicity of the displayed algorithm, we include points on the upper boundaries of ϱ and thus treat them as if they were in the inside. Then it suffices to compute the arrays $I^{(\alpha)}$ for all $\alpha \in \Delta$ and the case differentiation in the formula for $\mathcal{Y}_i$ and $\bar{\mathcal{Y}}_i$ is easier to handle.

We first give a procedure to compute the matrices I and I^α for each $\alpha \in \Delta$ which will be used in the algorithm. In the following we will call these matrices the cumulative sum arrays. As in subsection 2.1.3, we use some abstract data types for the passing of parameters in the algorithms. Again we make use of an abstract reference object REF of generic type OBJECT for accessing global data, such as the cumulative sum arrays, within different procedures.

Procedure `VerticalSum`($z \in \mathbb{R}^S$, $N_1, N_2 \in \mathbb{N}$, REF: *OBJECT*)

input : data $z \in \mathbb{R}^S$, dimensions $N_1, N_2 \in \mathbb{N}$ of S
output: array I holding vertical sums of values of z, I and dimensions N_1, N_2 are stored in REF

```
begin
    NEW(I, N1 + 1, N2 + 1);
    for x ← 0 to N1 do
        I(x,0) ← 0;
    end
    for y ← 1 to N2 do
        I(0,y) ← 0;
        for x ← 1 to N1 do
            I(x,y) ← z(x,y) + I(x,y−1);
        end
    end
    REF.I ← I; REF.N1 ← N1; REF.N2 ← N2;
end
```

Provided that the vertical sum array I has been computed with procedure VerticalSum, the following procedure can be used to compute the arrays $I^{(\alpha)}$ and $J^{(\alpha)}$ for a given angle $\alpha \in (-\pi/2, \pi/2)$. Note that the set K of numbers $k \in \mathbb{Z}$ such that $G_{k,\alpha}$ intersects the rectangle $[0, N_1] \times [0, N_2]$ is given as follows (for a detailed derivation compare Proof of Lemma 4.2.1). If $\sin\alpha \geq 0$, then

$$K = \left\{ k \in \mathbb{Z} : \frac{-N_1 \sin\alpha}{\max\{\cos\alpha, \sin\alpha\}} \leq (k + \frac{1}{2}) \leq \frac{N_2 \cos\alpha}{\max\{\cos\alpha, \sin\alpha\}} \right\},$$

and if $\sin\alpha < 0$, then

$$K = \left\{ k \in \mathbb{Z} : 0 \leq (k + \frac{1}{2}) \leq \frac{N_2 \cos\alpha - N_1 \sin\alpha}{\max\{\cos\alpha, |\sin\alpha|\}} \right\}.$$

Let now k be such that $G_{k,\alpha}$ intersects the rectangle $[0, N_1] \times [0, N_2]$. Let $\delta = \max\{|\cos\alpha|, |\sin\alpha|\}$ Denote the set of numbers $x \in \mathbb{Z}$ such that $G_{k,\alpha}(x) \in [0, N_1] \times [0, N_2]$ by R_k, then $R_k = [0, N_1] \cap \mathbb{Z}$, if $\sin\alpha = 0$,

$$R_k = \left\{ x \in [0, N_1] \cap \mathbb{Z} : -(k + \frac{1}{2})\frac{\delta}{\sin\alpha} \leq x \leq \frac{N_2 \cos\alpha - (k + \frac{1}{2})\delta}{\sin\alpha} \right\},$$

if $\sin\alpha > 0$, and

$$R_k = \left\{ x \in [0, N_1] \cap \mathbb{Z} : -(k + \frac{1}{2})\frac{\delta}{\sin\alpha} \geq x \geq \frac{N_2 \cos\alpha - (k + \frac{1}{2})\delta}{\sin\alpha} \right\},$$

if $\sin\alpha < 0$.

Procedure `CumulativeSum`($\alpha \in (-\pi/2, \pi/2)$, REF: *OBJECT*)

input : array I holding vertical sums of values, dimensions N_1, N_2, both in REF
output: array $I^{(\alpha)}$ stored in REF

begin
  $I \leftarrow$ REF.I; $N_1 \leftarrow$ REF.N_1; $N_2 \leftarrow$ REF.N_2;
  NEW($I^{(\alpha)}, N_1 + 1, N_2 + 1$);
  $\delta \leftarrow \max\{|\sin\alpha|, |\cos\alpha|\}$;
  if $\sin\alpha \geq 0$ **then**
    $k_{min} \leftarrow \lceil -\frac{N_1 \sin\alpha}{\delta} - \frac{1}{2} \rceil$; $k_{max} \leftarrow \lfloor -\frac{N_2 \cos\alpha}{\delta} - \frac{1}{2} \rfloor$;
    (— run through all potentially intersecting lines —)
    for $k \leftarrow k_{min}$ ***to*** k_{max} **do**
      $x_{min} \leftarrow 0$; $x_{max} \leftarrow N_1$;
      if $\sin\alpha \neq 0$ **then**
        $x_{min} \leftarrow \max\{x_{min}, \lceil -(k + \frac{1}{2}) \frac{\delta}{\sin\alpha} \rceil\}$;
        $x_{max} \leftarrow \min\{x_{max}, \lfloor \frac{N_2 \cos\alpha}{\sin\alpha} - (k + \frac{1}{2}) \frac{\delta}{\sin\alpha} \rfloor\}$
      end
      $s \leftarrow 0$;
      (— run through all positions on the line and in the rect —)
      for $x \leftarrow x_{min}$ ***to*** x_{max} **do**
        $y \leftarrow \lfloor g_{k,\alpha}(x) \rfloor$; $s \leftarrow s + I_{x,y}$; $I^{(\alpha)}_{x,y} \leftarrow s$;
      end
    end
  else
    $k_{min} \leftarrow 0$; $k_{max} \leftarrow \lfloor \frac{N_2 \cos\alpha - N_1 \sin\alpha}{\delta} - \frac{1}{2} \rfloor$;
    (— run through all potentially intersecting lines —)
    for $k \leftarrow k_{min}$ ***to*** k_{max} **do**
      $x_{min} \leftarrow \max\{0, \lceil \frac{N_2 \cos\alpha}{\sin\alpha} - (k + \frac{1}{2}) \frac{\delta}{\sin\alpha} \rceil\}$;
      $x_{max} \leftarrow \min\{N_1, \lfloor -(k + \frac{1}{2}) \frac{\delta}{\sin\alpha} \rfloor\}$;
      $s \leftarrow 0$;
      (— run through all positions on the line and in the rect —)
      for $x \leftarrow x_{min}$ ***to*** x_{max} **do**
        $y \leftarrow \lfloor g_{k,\alpha}(x) \rfloor$; $s \leftarrow s + I_{x,y}$; $I^{(\alpha)}_{x,y} \leftarrow s$;
      end
    end
  end
  REF.$I^{(\alpha)} \leftarrow I^{(\alpha)}$;
end

The following algorithm computes the sum of values of the array z in the inside plus the upper boundary of a polygon $\varrho \in \Lambda_n(\Delta, R)$ provided that for each $\alpha \in \Delta$ the array $I^{(\alpha)}$ has been computed with the previous procedure.

Algorithm 3.4: Determining the sum over the inside of a polygon

input : cumulative sum arrays I and I^{α} for each $\alpha \in \Delta$ in REF, $n \in \mathbb{N}$ and polygon $\varrho \in \Lambda_n(\Delta, R)$ specified by points $p_i \in \mathbb{R}^2$, $1 \le i \le n$

output: sum Z of values of array corresponding to I and $I^{(\alpha)}$ in the inside plus upper boundary of $\varrho \in \Lambda_n(\Delta, R)$

begin

$p_0 \leftarrow p_n$; $p_{n+1} \leftarrow p_1$;

$Z \leftarrow 0$;

for $k \leftarrow 0$ ***to*** n **do**

$x_1 \leftarrow p_k^{(1)}$; $x_2 \leftarrow p_{k+1}^{(1)}$;

(— check if vertical line —)

if $x_1 \neq x_2$ **then**

$x_0 \leftarrow p_{k-1}^{(1)}$; $y_0 \leftarrow p_{k-1}^{(2)}$; $y_1 \leftarrow p_k^{(2)}$; $y_2 \leftarrow p_{k+1}^{(2)}$;

if $x_1 < x_2$ **then**

$a \leftarrow x_1; b \leftarrow x_2; y_1' \leftarrow x_1; y_2' \leftarrow x_2$;

else

$a \leftarrow x_2; b \leftarrow x_1; y_1' \leftarrow x_2; y_2' \leftarrow x_1$;

end

(— determine line parameters —)

$\alpha = \arctan \frac{y_2' - y_1'}{b-a}$; (*if not known a priori*)

$k \leftarrow (ay_2' - by_1')/\max\{b-a, |y_2' - y_1'|\}$; (* dito *)

(— determine corresponding integer positions —)

$X_1 \leftarrow \lfloor a \rfloor$; $Y_1 \leftarrow \lfloor g_{k,\alpha}(\lfloor a \rfloor) \rfloor$; $X_2 \leftarrow \lceil b-1 \rceil$; $Y_2 \leftarrow \lfloor g_{k,\alpha}(\lceil b-1 \rceil) \rfloor$;

— Y_2 in range ? —

if $X_2 \neq X_1$ **then**

(— Y_1 in range ? —)

if $0 \le Y_1 \le N_2$ **then**

$Z \leftarrow I^{(\alpha)}_{(X_2,Y_2)} - I^{(\alpha)}_{(X_1,Y_1)}$

else

$Z \leftarrow I^{(\alpha)}_{(X_2,Y_2)}$

end

(— left-right or right-left ? —)

if $X_1 < X_2$ **then**

$s \leftarrow s + Z$

else

$s \leftarrow s - Z$

end

end

(— now check if machine-arithmetically $x_1 \in \mathbb{Z}$ —)

if $\lceil x_1 \rceil = \lfloor x_1 \rfloor$ **then**

(— maintains direction ? —)

if $x_0 < x_1 < x_2$ **then** $s \leftarrow s + I_{(X_1,Y_1)}$;

else if $x_0 > x_1 > x_2$ **then** $s \leftarrow s - I_{(X_1,Y_1)}$ **end** ;

end

end

end

end

Remark 3.3.28: *In Algorithm 3.4 and Procedure CumulativeSum, floating point computations are performed to determine the correct positions on the arrays for the computation of the cumulative sums corresponding to a polygon. To each polygon specified by edges in the plane* $\mathbb{R}^2$ *there corresponds a* digital *polygon, a subset of digital lines in the discrete domain* $\mathbb{Z}^2$*. Conversely, to each digital polygon there is some set of continuous lines that correspond to the digital edges. However, the summation trick developed in the proof of Lemma 3.3.20 and in that of Theorem 3.3.27 can not be carried forward to digitally specified polygons in a straightforward way. Digital lines with different directions do not necessarily intersect and – even worse – two different digital lines might have more than one point in common. Assume we want to compute the sum of values of* z *in the inside of a digital polygon. If digital edges from left to right are understood as upper boundary and lines from right to left as lower boundary, then they correspond to (not unique) continuous edges that are located directly under and over the digital lines, respectively. But then in general the continuous edges do not form a polygon, compare Figure 3.29. The same occurs if the respective computation for the inside plus boundary is performed. If upper edges are contained but lower ones are not, then still the corresponding continuous edges do not necessarily form a polygon, but interestingly, on the 'problematic' regions near the vertices the differences between the lower and upper boundaries vanish.*

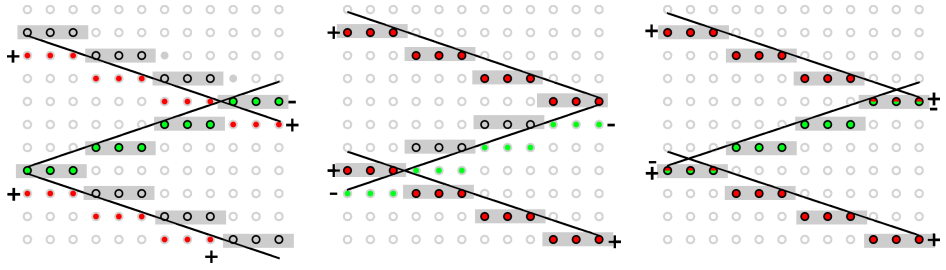


Fig. 3.29: *Situations occurring when the summation trick is applied on digital edges. Left: upper and lower boundary not included. Middle: upper and lower edges included. Right: Upper boundary included, lower boundary not included. Black lines are corresponding continuous edges.*

We claim that for each digital polygon there is some continuous polygon such that the latter summation scheme corresponds to the continuous case. This would mean that Algorithm 3.4 could be modified such that it produced the same result as it does now while exclusively using integer arithmetics. For

special cases such as the wedge intersection of a rectangle, this can be easily shown to be the case. Algorithms for the relevant cases of this thesis have therefore been implemented using integer arithmetics. There basically for each angle a reference line is stored, as used and computed in Procedure CumulativeSum, and the values corresponding to wedge intersections are then computed using shifted versions of this reference lines.

3.4 Synopsis

In this chapter we have provided what is necessary to perform a minimization of Potts functionals on wedge divided rectangles.

Section 3.1 was devoted to the minimization of the functional $\sum_{s\in r} \varrho(z_s - f_r(s))$ over $f_r \in \mathcal{F}_r$. In Subsection 3.1.3 it was shown that an efficient computation of the projection $\Pi_{\mathcal{F}_r} z_r$ and the approximation error $\|z_r - \Pi_{\mathcal{F}_r} z_r\|_2^2$ is possible, if $\mathcal{F}_r$ is a linear function space spanned by a finite family of functions $\varphi_i \in \mathbb{R}^r$, $1 \le i \le n$, $(n \in \mathbb{N})$, and if the $\frac{(n+1)(n+2)}{2}$ 'moments' $\sum_{s\in r} \varphi_i(s)\varphi_j(s)$, $\sum_{s\in r} \varphi_i(s) z_s$ and $\sum_{s\in r} z_s^2$, $1 \le i, j \le n$, can be computed rapidly.

In Section 3.2 we commented on the partition of rectangles by intersection with a line, into a partition of wedges. To get the link between wedges on the continuous domain $[0,1)^2$, and on the discrete domain $S = \{1,\dots,N_1\} \times \{1,\dots,N_2\}$, $N_1, N_2 \in \mathbb{N}$, a short survey on digital lines was given. In Subsection 3.2.3 we demonstrated that each linear dichotomy on the discrete domain corresponds to one and only one pair of points in S and therefore an estimation of the number of wedges could be given. Additionally, an algorithm was derived for an efficient determination of the pair of points corresponding to a linear dichotomy.

In Section 3.3 we developed an algorithm for efficient regression over polygonal domain. We have started with results concerning the intersection of two Jordan curves over continuous domain. Utilizing these results, we have derived a rule to integrate a function over discrete domain within a continuous Jordan curve. We have carried the results forward to the discrete domain and have presented an algorithm to efficiently integrate a function over polygonal domain, provided that certain cumulative matrices have been computed beforehand.

Generally in this thesis we are interested in segmentations that allow an

efficient minimization of the Potts functional. In Chapter 2 we have presented two partition classes that allow a fast traversing. As already indicated in Paragraph 2.3 we are interested in regression on some comprehensive function space over each element of such a partition that is generated by an additional intersection of each fragment by a line. Now because wedges are of polygonal form, we can use the results and algorithms of Section 3.3 to rapidly compute the minimizers of the Potts functional for a certain segmentation class. This will be discussed in detail in the next chapter.

4. Efficient Minimization of Potts Functionals

In this chapter we will merge the results of Chapters 2 and 3 and derive the basic object of this thesis: A segmentation class for which an efficient minimization of the Potts functional is possible. For the development of a fast regression scheme we will make use of the recursion Algorithms 2.3 and 2.8 displayed in Subsections 2.1.3 and 2.2.3, and of the fast integration Algorithm 3.4 presented in Section 3.3.

Let $N_1, N_2 \in \mathbb{N}$. We take on the notation from Chapter 1 and let $S = \{1, \dots, N_1\} \times \{1, \dots, N_2\}$ and let $\mathcal{R}$ denote a set of subsets of S, $\mathfrak{P} = \mathfrak{P}(\mathcal{R})$ a set of partitions of S and $\mathcal{F} = (\mathcal{F}_r)_{r \in \mathcal{R}}$ a family of functions, $\mathcal{F}_r \in \mathbb{R}^r$ for each $r \in \mathcal{R}$. The symbol $\mathfrak{S} = \mathfrak{S}(\mathfrak{P}, \mathcal{F})$ stands for the family of segmentations that is induced by the partition class $\mathfrak{P}$ together with the class of admissible functions $\mathcal{F}$.

Recall that the Potts functional with parameter $\gamma \geq 0$ is the mapping

$$\begin{aligned} &H_\gamma(\cdot, \cdot) : \mathbb{R}^S \times \mathfrak{S} \longrightarrow \mathbb{R} \\ &H(z, (\mathcal{P}, f_\mathcal{P})) = \gamma|\mathcal{P}| + \|z - f_\mathcal{P}\|_2^2. \end{aligned}$$

In Section 1.3 we have presented a generic minimization algorithm for the Potts functional. The structure of this algorithm demonstrates an application of the reduction principle. But it has a very large time complexity depending on the class of fragments and the function spaces, and is therefore unusable in realistic situations. Moreover, this algorithm does not contain a strategy to treat the local regression problem, and does not tell what kind of partitions are well suited for applications. We will now introduce a class of segmentations that allow an efficient minimization of the Potts functional.

4.1 Wedge Segmentations

In Chapter 2 two different partition classes have been displayed that – by different kinds of recursive structure – allow an efficient enumeration of its

elements. Both classes contain partitions that exclusively consist of rectangles. We presented algorithms that efficiently traverse the class of partitions for the purpose of minimization of the Potts functional. The problem of local minimization of $\|z|_r - f_r\|_2^2$ over $f_r \in \mathcal{F}_r$ for each fragment r had been left for Chapter 3. There it has been shown that for a linear function space spanned by a finite family of functions $\varphi_i \in \mathbb{R}^r$, $1 \leq i \leq n$, $(n \in \mathbb{N})$ an efficient computation of the local minimizers is possible if the $\frac{(n+1)(n+2)}{2}$ 'moments' $\sum_{s\in r} \varphi_i(s)\varphi_j(s)$, $\sum_{s\in r} \varphi_i(s)z_s$ and $\sum_{s\in r} z_s^2$, $1 \leq i, j \leq n$, could be computed rapidly. This in turn has been proved to be the case if r is a polygonal domain for a polygon ϱ stemming from the class $\Lambda(\Delta, R)$ with a finite set of angles $\Delta \subset (-\pi/2, \pi/2]$. For that reason we will now consider **wedge segmentations**, two-dimensional dyadic or hierarchic partitions where the fragments additionally are divided by intersections with lines. But in order that the fast integration algorithm can be applied we impose certain restrictions on the set of admitted lines.

Let $k \in \mathbb{Z}$ and $\alpha \in (-\pi, \pi]$ and a rectangle $r \subset S$ be given. Consider the wedge division $\mathcal{T}_{k,\alpha}$ of r given with $\delta_\alpha = \max\{|\sin\alpha|, |\cos\alpha|\}$ by the two sets

$$A_{k,\alpha}(r) := \{(x, y) \in r : x \sin\alpha + \delta_\alpha(k + \frac{1}{2}) > y\cos\alpha\}, \tag{4.1}$$

$$B_{k,\alpha}(r) := \{(x, y) \in r : x \sin\alpha + \delta_\alpha(k + \frac{1}{2}) \leq y\cos\alpha\}, \tag{4.2}$$

i.e. $\mathcal{T}_{k,\alpha} := \{A_{k,\alpha}(r), B_{k,\alpha}(r)\}$. We are now ready for the following

Definition 4.1.1 (Dyadic/Hierarchic Wedge Partition): *Let $\mathcal{P}$ be a dyadic or hierarchic partition of S and assume a finite set of angles $\Delta \subset (-\pi/2, \pi/2]$ is given. For each $r \in \mathcal{R}$ let $k_r \in \mathbb{Z}$ and $\alpha_r \in \Delta$ be given. A partition $\mathcal{W}$ of the form*

$$\mathcal{W} = \bigcup_{r\in\mathcal{P}} \mathcal{T}_{k_r,\alpha_r} \setminus \emptyset$$

will be called a **dyadic** *or* **hierarchic wedge partition**, *respectively, with angles Δ.*

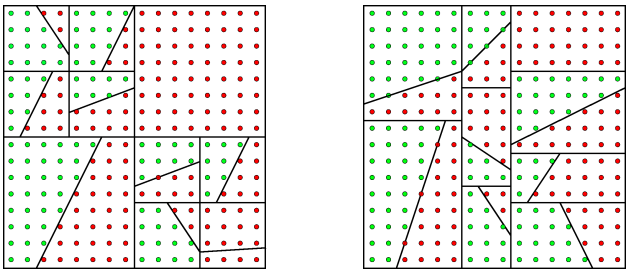


Fig. 4.1: Dyadic (left) and hierarchic (right) wedge partition.

We will now define the main objects of this thesis. Recall that for a function $f \in \mathbb{R}^S$ and a subset $r \subset S$, the symbol $f|_r$ denotes the function f restricted to r.

Definition 4.1.2 (Dyadic/Hierarchic Wedge Segmentation): *Let $\mathcal{W}$ be a dyadic or hierarchic wedge partition and a function space $\mathcal{F} \subset \mathbb{R}^S$ be given. For each $r \in \mathcal{R}$ consider the function space defined by $\mathcal{F}_r = \operatorname{span}\{f|_r : f \in \mathcal{F}\}$ and let $f_r \in \mathcal{F}_r$. A segmentation of the form*

$$(\mathcal{P}, f_{\mathcal{P}}) = (\mathcal{P}, (f_r)_{r\in\mathcal{P}})$$

will then be called a **dyadic** *or* **hierarchic wedge $\mathcal{F}-$segmentation**, *respectively, with angles Δ and function space $\mathcal{F} = \operatorname{span}\{\varphi_i : 1 \le i \le n\}$.*

In the following we denote the class of dyadic and hierarchic wedge segmentations with angles Δ and function space $\mathcal{F}$ by $\mathfrak{W}^d(\Delta, \mathcal{F})$ and $\mathfrak{W}^h(\Delta, \mathcal{F})$, respectively.

4.2 Traversing the Wedges and Local Models

In Paragraphs 2.1.3 and 2.2.3 we showed how to efficiently traverse the class of dyadic and hierarchic partitions. We will now extend the corresponding recursion schemes to the wedge segmentations defined in the previous section. We assume the set $\Delta \subset (-\pi/2, \pi/2]$ to be finite.

For $\alpha \in (-\pi, \pi]$ and $k \in \mathbb{Z}$ recall the notation

$$G_{k,\alpha} = \{(x, y) \in \mathbb{R}^2 : y \cos\alpha = (k + \frac{1}{2}) \max\{|\cos\alpha|, |\sin\alpha|\} + x \sin\alpha\}.$$

The number of such lines intersecting a rectangle can be estimated by the following

Lemma 4.2.1: *Let $\alpha \in [-\pi/2, \pi/2)$ and a rectangle $R = [l, r]\times[b, t]$, $l, r, b, t \in \mathbb{R}$, $l < r$, $b < t$, be given. Then*

$$|\{k \in \mathbb{Z} : G_{k,\alpha} \cap R \neq \emptyset\}| \leq (r - l) + (t - b)$$

Proof. Let $K = \{k \in \mathbb{Z} : G_{k,\alpha} \cap R \neq \emptyset\}$. Then

$$K = \bigcup_{(x,y)\in R} \left\{ k \in \mathbb{Z} : (k + \frac{1}{2}) = \frac{y \cos\alpha - x \sin\alpha}{\max\{\cos\alpha, |\sin\alpha|\}} \right\}$$

The assumption $\alpha \in [-\pi/2, \pi/2)$ implies $\cos\alpha \geq 0$. Let additionally $\sin\alpha \geq 0$. Then

$$K = \left\{ k \in \mathbb{Z} : \frac{b\cos\alpha - r\sin\alpha}{\max\{\cos\alpha, \sin\alpha\}} \leq (k + \frac{1}{2}) \leq \frac{t\cos\alpha - l\sin\alpha}{\max\{\cos\alpha, \sin\alpha\}} \right\}. \quad (4.3)$$

Therefore

$$|K| \leq \frac{(t - b)\cos\alpha + (r - l)\sin\alpha}{\max\{\cos\alpha, \sin\alpha\}} \leq (t - b) + (r - l).$$

If $\sin\alpha < 0$, then

$$K = \left\{ k \in \mathbb{Z} : \frac{b\cos\alpha - l\sin\alpha}{\max\{\cos\alpha, |\sin\alpha|\}} \leq (k + \frac{1}{2}) \leq \frac{t\cos\alpha - r\sin\alpha}{\max\{\cos\alpha, |\sin\alpha|\}} \right\} \quad (4.4)$$

and thus

$$|K| \leq \frac{(t - b)\cos\alpha + (l - r)\sin\alpha}{\max\{\cos\alpha, |\sin\alpha|\}} \leq (t - b) + (r - l).$$

□

For two functions $f, g \in \mathbb{R}^S$ we let in the following $f \cdot g : S \to \mathbb{R}$, $(f \cdot g)(s) = f(s) \cdot g(s)$. Moreover, we identify data $z \in \mathbb{R}^S$ with the mapping $z : S \to \mathbb{R}$. Recall the definitions for symbols $A_{k,\alpha}$ and $B_{k,\alpha}$ on page 114 and the definition of the cumulative sum arrays in Notations 3.3.18 and 3.3.24.

Lemma 4.2.2: *Consider a function space $\mathcal{F} \subset \mathbb{R}^S$ that is spanned by a family of $n \in \mathbb{N}$ functions $\varphi_i \in \mathbb{R}^S$, $1 \leq i \leq n$. Let an angle $\alpha \in (-\pi/2, \pi/2]$ and a rectangle $R \subset \mathbb{R}^2$ with side lengths $w > 0$ and $h > 0$ be given. Assume $F = \{\varphi_i \cdot \varphi_j : 1 \leq i, j \leq n\} \cup \{\varphi_i \cdot z : 1 \leq i \leq n\} \cup \{z^2\}$ and suppose that for each $f \in F$ the cumulative sums $I(f)$, $I^{(\alpha)}(f)$ and $J^{(\alpha)}(f)$ have already*

been computed.
Then the minimization problem

$$(k,\lambda,\mu) \longrightarrow \min_{k\in\mathbb{Z},\lambda,\mu\in\mathbb{R}^n} \|\mathbb{1}_{A_{k,\alpha}(R)}(z-\sum_{i=1}^{n}\lambda_i\varphi_i)\|_2^2 + \|\mathbb{1}_{B_{k,\alpha}(R)}(z-\sum_{i=1}^{n}\mu_i\varphi_i)\|_2^2$$

can be solved with a time complexity of $O((w+h)\cdot n^3)$ *and a space complexity of* $O(n^2)$.

Proof. By Lemma 4.2.1 the number of integers k such that the rectangle R has non-empty intersection with the line $G_{k,\alpha}$ is less or equal $w+h$. In the remaining cases the line has empty intersection with R and does therefore induce the segmentation $\mathcal{T}_{k,\alpha}(R)=\{R\}$. Let $\Delta=\{0,\pi/2,\alpha\}$. Because the sets $A_{k,\alpha}(R)$ and $B_{k,\alpha}(R)$ are specified by bordering polygons ϱ with 3, 4 or 5 vertices on lines $G_{k,\beta}$ with $k\in\mathbb{Z}$ and $\beta\in\Delta$, they are elements of the sets $\Lambda_i(\Delta,[0,N_1]\times[0,N_2])$. By Lemma 3.3.26, for each function $f\in\mathbb{R}^S$ the values $\sum_{s\in I(\varrho)} f(s)$ can be determined with a time complexity of $O(5)=O(1)$ and a space complexity of $O(1)$. By Lemma 3.1.13, for fixed $k\in\mathbb{Z}$ the minimization of $\|\mathbb{1}_{A_{k,\alpha}(R)}(z-\sum_{i=1}^{n}\lambda_i\varphi_i)\|_2^2$ in λ is possible with a time complexity of $O(n^3)$ and a space complexity of $O(n^2)$. The same holds for the minimization of $\|\mathbb{1}_{B_{k,\alpha}(R)}(z-\sum_{i=1}^{n}\lambda_i\varphi_i)\|_2^2$. The traversing of all wedges in R implies a maximal factor of $(w+h)$. □

Now we present an algorithm to do the local minimization of the Potts functional, as displayed in Lemma 4.2.2. Therefore we need to know where each line intersects the rectangle $R=[l,r]\times[b,t]$. Let $\alpha\in[-\pi/2,\pi/2)$. First, the set of integers $k\in\mathbb{Z}$ such that line $G_{k,\alpha}$ intersects rectangle R is given by equation (4.3) for $\sin\alpha\geq 0$ and by (4.4) for $\sin\alpha<0$. Provided that $g_{k,\alpha}\cap R\neq\emptyset$, the intersection parameters $x_1,x_2\in\mathbb{R}^2$ with $(x_1,g_{k,\alpha}(x_1))\in\partial(R)$, $(x_2,g_{k,\alpha}(x_2))\in\partial(R)$ are given with $\delta_k:=(k+\frac{1}{2})\max\{|\cos\alpha|,|\sin\alpha|\}$ by

$$(x_1,x_2)=\begin{cases}(l,r) & \text{if } \sin\alpha=0,\\ (k+\frac{1}{2},k+\frac{1}{2}) & \text{if } \sin\alpha=1,\\ (\max\{l,\frac{b\cdot\cos\alpha-\delta_k}{\sin\alpha}\},\min\{r,\frac{t\cdot\cos\alpha-\delta_k}{\sin\alpha}\}) & \text{if } \sin\alpha>0,\\ (\max\{l,\frac{t\cdot\cos\alpha-\delta_k}{\sin\alpha}\},\min\{r,\frac{b\cdot\cos\alpha-\delta_k}{\sin\alpha}\}) & \text{if } \sin\alpha<0.\end{cases}$$

The vertical coordinates of the intersection points are given by

$$(y_1,y_2)\leftarrow\begin{cases}(b,t) & \text{if } \sin\alpha=1,\\ (g_{k,\alpha}(x_1),g_{k,\alpha(x_2)}) & \text{otherwise.}\end{cases}$$

The intersection points determine two simple polygons that have the line from (x_1, y_1) to (x_2, y_2) in common. These are the boundaries of the wedges $A_{k,\alpha}(R)$ and $B_{k,\alpha}(R)$.

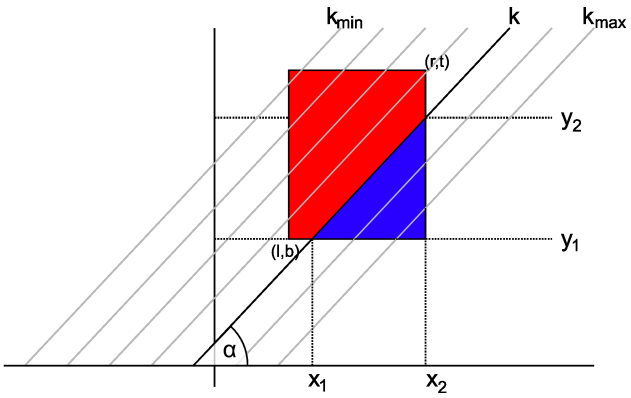


Fig. 4.2: Different wedges $A_{k,\alpha}(R)$ and $B_{k,\alpha}(R)$ for fixed angle α.

Algorithm 4.1: Minimize $(k, \lambda, \mu) \longrightarrow \|\mathbb{1}_{A_{k,\alpha}(R)}(z - \sum_{i=1}^{n} \lambda_i \varphi_i)\|_2^2 + \|\mathbb{1}_{B_{k,\alpha}(R)}(z - \sum_{i=1}^{n} \mu_i \varphi_i)\|_2^2$

input : $n \in \mathbb{N}$,$\alpha \in (-\pi/2, \pi/2]$, cumulative sum arrays $I(f)$ and $I^{\alpha}(f)$ and $I^{(0)}(f)$ for all $f \in \{\varphi_i \cdot \varphi_j : 1 \le i, j \le n\} \cup \{\varphi_i \cdot z : 1 \le i \le n\} \cup \{z^2\}$, rectangle $R = [l, r] \times [b, t]$

output: extremal points $\hat{k} \in \mathbb{Z}$, $\lambda, \mu \in \mathbb{R}^n$ and approximation error m

begin

(— determine intersecting lines —)

$\delta \leftarrow \max\{\cos\alpha, \sin\alpha\}$

$$(k_{min}, k_{max}) \leftarrow \begin{cases} (\lceil l \rceil, \lfloor r \rfloor) & \text{if } \sin\alpha = 1 \\ (\frac{b\cos\alpha - r\sin\alpha}{\delta} - \frac{1}{2}, \frac{t\cos\alpha - l\sin\alpha}{\delta} - \frac{1}{2}) & \text{if } 0 \le \sin\alpha < 1\,; \\ (\frac{b\cos\alpha - l\sin\alpha}{\delta} - \frac{1}{2}, \frac{t\cos\alpha - r\sin\alpha}{\delta}) & \text{if } \sin\alpha < 0 \end{cases}$$

$m \leftarrow \infty$;

for $k \leftarrow k_{min}$ ***to*** k_{max} **do**

$\delta_k \leftarrow (k + \frac{1}{2})\delta$;

(— determine intersection points —)

$$(x_1, x_2) \leftarrow \begin{cases} (l, r) & \text{if } \sin\alpha = 0, \\ (k + \frac{1}{2}, k + \frac{1}{2}) & \text{if } \sin\alpha = 1 \\ (\max\{l, \frac{b\cdot\cos\alpha - \delta_k}{\sin\alpha}\}, \min\{r, \frac{t\cdot\cos\alpha - \delta_k}{\sin\alpha}\}) & \text{if } 0 < \sin\alpha < 1, \\ (\max\{l, \frac{t\cdot\cos\alpha - \delta_k}{\sin\alpha}\}, \min\{r, \frac{b\cdot\cos\alpha - \delta_k}{\sin\alpha}\}) & \text{if } \sin\alpha < 0. \end{cases};$$

$$(y_1, y_2) \leftarrow \begin{cases} (b, t) & \text{if } \sin\alpha = 1 \\ (g_{k,\alpha}(x_1), g_{k,\alpha(x_2)}) & \text{otherwise} \end{cases};$$

determine polygons ϱ and ϱ' corresponding to the rectangle r and intersection points (x_1, y_1) and (x_2, y_2).

(— compute all 'moments' —)

foreach $f \in \{\varphi_i \cdot \varphi_j : 1 \le i, j \le n\} \cup \{\varphi_i \cdot z : 1 \le i \le n\} \cup \{z^2\}$ **do**

call Algorithm 3.4 with ϱ and ϱ' and f;

Store result in $A_{f,\varrho}$ and $A_{f,\varrho'}$

end

(— set up linear equation system (Subsection 3.1.3) —)

fill arrays $M := (A_{\varphi_i\cdot\varphi_j,\varrho})_{1\le i,j\le n}$, $Y := (A_{z_s\cdot\varphi_j(s),\varrho})_{1\le j\le n}$, $M' := (A_{\varphi_i\cdot\varphi_j,\varrho'})_{1\le i,j\le n}$ and $Y' := (A_{z_s\cdot\varphi_j(s),\varrho'})_{1\le j\le n}$;

(— solve linear equation system , Trefethen and Bau (1997), pp. 83-85 —)

compute solutions $\hat{a}$ and $\hat{b}$ of $Y = M \cdot a$ and $Y' = M' \cdot b$;

(— compute approximation error (Subsection 3.1.3) —)

$m_1 \leftarrow A_{z_s^2,\varrho} - 2\sum_{i=1}^{n} \hat{a}_i Y_i + \sum_{i=1}^{n} \hat{a}_i^2 M_{ii}$;

$m_2 \leftarrow A'_{z_s^2,\varrho'} - 2\sum_{i=1}^{n} \hat{b}_i Y'_i + \sum_{i=1}^{n} \hat{b}_i^2 M'_{ii}$;

(— check if minimal —)

if $m_1 + m_2 < m$ **then**

$m \leftarrow m_1 + m_2$;

$\lambda \leftarrow \hat{a}$; $\mu \leftarrow \hat{a}'$;

$\hat{k} \leftarrow k$;

end

end

end

4.3 The Main Theorem and Algorithm

We can now present a central result of this thesis. It embraces the main statements of the Chapters 2 and 3.

Theorem 4.3.1: *Consider a function space $\mathcal{F} \subset \mathbb{R}^S$ that is spanned by a family of $n \in \mathbb{N}$ functions $\varphi_i \in \mathbb{R}^S$, $1 \leq i \leq n$. Let $\Delta \subset (-\pi/2, \pi/2]$ be finite and $z \in \mathbb{R}^S$. The Potts functional*

$$H(z, (\mathcal{P}, f_{\mathcal{P}})) = \gamma|\mathcal{P}| + \|z - f_{\mathcal{P}}\|_2^2$$

can be minimized with

1. *a time complexity of $O(|S| \cdot |\Delta| \cdot n^3)$ and a spatial complexity of $O(|\Delta| \cdot |S| \cdot n^2)$ over the class of dyadic wedge $\mathcal{F}$-segmentations with angles Δ,*
2. *a time complexity of $O(|S|^2(N_1 + N_2) \cdot |\Delta| \cdot n^3)$ and a spatial complexity of $O(|\Delta| \cdot |S| \cdot n) \cdot n^2$ over the class of hierarchic wedge $\mathcal{F}$-segmentations with angles Δ.*

Proof. By Lemma 4.2.2, the local minimization complexity for each rectangular fragment and each angle $\alpha \in \Delta$ is given by $O((w + h) \cdot n^3)$. Corollary 2.2.19 states that the global minimization has a complexity of $O(|S|c_1)$ if the local minimization complexity for a rectangle with side lengths w and h is given by $O(c_1(w + h))$. Because the local minimization can be performed by successively comparing best results for different angles $\alpha \in \Delta$, we get a global minimization complexity of $O(|S| \cdot |\Delta| \cdot n^3)$. For the hierarchic partitions we use Corollary 2.1.8 and get a global minimization complexity of $O(|S|^2(N_1 + N_2) \cdot |\Delta| \cdot n^3)$. In both cases, by Lemma 4.2.2 the space complexity for the local minimization process is for each angle $\alpha \in \Delta$ given by $O(|S| \cdot n^2)$, and therefore a global memory of $O(|S| \cdot n^2 \cdot |\Delta|)$ has to be allocated. For the reconstruction part of Algorithms 2.3 and 2.8 there is a global time complexity of $O(|S| \cdot n)$ because for each rectangle R the approximation can be done by evaluating the sum $\sum_{i=1}^{n} \varphi_i(s)$ for each s in R and it is therefore linear and thus Corollaries 2.1.8 and 2.2.19 can be applied. □

Now we display the algorithm to do the minimization of the Potts model for the hierarchic and the dyadic wedge segmentations.

Algorithm 4.2: Minimize the Potts functional over hierarchic or dyadic wedge partitions

input : $S = \{1, \dots, N_1\} \times \{1, \dots, N_2\}$, $\Delta \subset (-\pi/2, \pi/2]$ with $|\Delta| = n$, Functions φ_i, $1 \le i \le n$, data $z \in \mathbb{R}^S$

output: Minimizer $M = (M_s)_{s \in S}$ of the Potts functional

begin

(— check that 0 degree is contained in the set of angles —)

$\Delta \leftarrow \Delta \cup \{0\}$;

(— create the result matrix —)

NEW(M, N_1, N_1);

(— create, compute and store cumulative matrices —)

foreach $f \in \{\varphi_i \cdot \varphi_j : 1 \le i, j \le n\} \cup \{\varphi_i \cdot z : 1 \le i \le n\} \cup \{z^2\}$ **do**

VerticalSum$(z, N_1, N_2,$ REF$)$;

$I(f) \leftarrow$ REF.I;

foreach $\alpha \in \Delta$ **do**

CumulativeSum$(\alpha,$ REF$)$;

$I^{(\alpha)}(f) \leftarrow$ REF.I;

end

end

Call Algorithm 2.3 or Algorithm 2.8 with the following local minimization and approximation rules: ;

Computation of $\|\Pi_{\mathcal{F}_{[l,r]\times[b,t]}} z - z_{[l,r]\times[b,t]}\|_2^2$:

begin

(— local minimization rule, known: rectangle $R = [l, r] \times [b, t]$ —)

$\hat{m} \leftarrow \infty$;

foreach $\alpha \in \Delta$ **do**

Call Algorithm 4.1 with n, α, φ_i $(1 \le i \le n)$, $I^{\alpha}(f)$, $I^{(0)}(f)$ and $I(f)$ for all $f \in \{\varphi_i \cdot \varphi_j : 1 \le i, j \le n\} \cup \{\varphi_i \cdot z : 1 \le i \le n\} \cup \{z^2\}$ and $R' = [l - \frac{1}{2}, r + \frac{1}{2}] \times [b - \frac{1}{2}, t + \frac{1}{2}]$. Result: $\lambda, \mu, \hat{k}, m$;

if $M > m$ **then**

$\hat{m} \leftarrow m$; $\hat{\alpha} \leftarrow \alpha$; $\hat{\lambda} \leftarrow \lambda$; $\hat{\mu} \leftarrow \mu$; $\hat{K} \leftarrow \hat{k}$;

end

end

Return values $\hat{m}$, $\hat{\alpha}$, $\hat{\lambda}$, $\hat{\mu}$, $\hat{K}$;

end

Computation $M_{[l,r]\times[b,t]} \leftarrow \Pi_{\mathcal{F}_{[l,r]\times[b,t]}} z$:

begin

(— local approximation rule, known: rectangle $R = [l, r] \times [b, t]$, $\hat{m}$, $\hat{\alpha}$, $\hat{\lambda}$, $\hat{\mu}$, $\hat{K}$ —)

for $x \leftarrow l$ ***to*** r **do**

for $y \leftarrow r$ ***to*** r **do**

if $x \sin\hat{\alpha} + (\hat{K} + \frac{1}{2}) \cdot \max\{|\sin\hat{\alpha}|, \cos\hat{\alpha}\} > y \cos\hat{\alpha}$ **then**

(— In $A_{\hat{K},\alpha}(R)$—)

$M_{x,y} \leftarrow 0$;

for $i \leftarrow 1$ ***to*** n **do**

$M_{x,y} \leftarrow M_{x,y} + \hat{\lambda}_i \varphi_i(x, y)$

end

else

(— In $B_{\hat{K},\alpha}(R)$—)

$M_{x,y} \leftarrow 0$ **for** $i \leftarrow 1$ ***to*** n **do**

$M_{x,y} \leftarrow M_{x,y} + \hat{\mu}_i \varphi_i(x, y)$

end

end

end

end

end

end

Remark 4.3.2: *Since for a dyadic partition, minimization results can be stored in the quad tree with a space complexity of $O(|S| \cdot n)$, the algorithm can also be formulated such that a memory consumption of $O(|S| \cdot n)$ is used. Then the local minimizers are computed successively for each angle over the whole quad tree, where at each fragment the 'current' minimization result is compared with the best previous one. Another, even more important, point is this memorizing of the local minimization results for each rectangle in a tree can be used for a very fast access to minimizers of H_γ for different values of γ, compare the runtime discussion in Paragraph 6.2.2.*

5. Consistency

This chapter is devoted to the exploration of (wedge) segmentations resulting from minimizing Potts functionals with respect to consistency in a nonparametric regression model. The verification of consistency is often *the* mathematical justification for the application of estimation procedures to real data. Asymptotic features are sometimes also taken for lack of assertions in the finite case (often, for example, used in testing theory). Above all, asymptotic features, such as consistency, reveal basic structure behind the objects considered. Consistency is neither the beginning nor the end of an analysis of the studied procedure. It is rather a *minimal* requirement and provides the valuable indication that the estimation is in line with the 'truth' for large enough data sets.

In statistics one is additionally interested in statements about optimality. Concerning nonparametric regression this refers to the specification of the rate of convergence for estimators.

5.1 A Nonparametric Regression Model

In this section, we display what kind of data we consider for the analysis of consistency. We will start with the model and describe how we think that discrete data are collected from some continuous source. Then we will present discretization- and embedding-operators and derive some properties. Finally, we will specify postulations about the noise of the observed data.

5.1.1 Specification of the Model

We start with the following basic postulates:

- First, we assume that there is unrevealed 'truth' given as a real valued signal f over the continuous domain $[0,1)^2$. We look upon such a signal as an image which we additionally presume to stem from the subclass $L^2([0,1)^2)$.

- Second, we assume that images $f \in L^2([0,1)^2)$ are observed as erroneous measurements on rectangular $n \times n$ - grids over $[0,1)^2$.

More precisely, we consider in the sequel a grid size $n \in \mathbb{N}$, a function $f \in L^2((0,1]^2)$, a discrete index set $S_n := \{1,\dots,n\}^2$ and the following continuous $(\frac{1}{n} \times \frac{1}{n})$-cells:

$$I_{ij}^{(n)} := \left[\frac{i-1}{n}, \frac{i}{n}\right) \times \left[\frac{j-1}{n}, \frac{j}{n}\right), \quad i,j \in \mathbb{N}, 1 \le i,j \le n.$$

Moreover, let $\xi^{(n)} = (\xi_{ij}^{(n)})_{1\le i,j\le n}$ be a matrix of independent random variables. We will give an exact definition of the admitted random variables later on. A measurement $z^{(n)}$ of $f \in L^2([0,1)^2)$ is then given by a matrix $z^{(n)} \in \mathbb{R}^{S_n}$ with

$$z_{ij}^{(n)} = |S_n| \int_{I_{ij}^{(n)}} f(r)\, dr + \xi_{ij}^{(n)}, \quad 1 \le i,j \le n. \tag{5.1}$$

Thus, for us, a measurement is an averaging over the cells and the error is produced by the process of measurement.

In the following, we use the notation $f^{(n)} := (f_{ij}^{(n)})_{1\le i,j\le n}$ for the averaged version of f defined by

$$f_{ij}^{(n)} := |S_n| \int_{I_{ij}^{(n)}} f\, d\lambda, \quad 1 \le i,j \le n,$$

and can thus write shortly

$$z^{(n)} = f^{(n)} + \xi^{(n)} \tag{5.2}$$

for equation (5.1). Note, that $f^{(n)}$ is the conditional expectation of f with respect to the σ-field generated by the sets $I_{ij}^{(n)}$, i.e.

$$f^{(n)} = \mathbb{E}(f|\sigma(\{I_{ij}^{(n)}, 1 \le i,j \le n\})).$$

Recall that for a segmentation $(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})$ over S_n the squared Euclidean distance to $z^{(n)}$ is given by

$$\|z^{(n)} - g_{\mathcal{P}}^{(n)}\|_2^2 = \sum_{r\in\mathcal{P}} \sum_{s\in r} (z_s^{(n)} - g_r^{(n)}(s))^2.$$

In this chapter we will consider a Potts functional with parameter $\gamma > 0$ where the data term is additionally weighted with $|S_n|^{-1}$, i.e.

$$H_\gamma^{(n)}(z^{(n)}, (\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})) = \gamma|\mathcal{P}^{(n)}| + \frac{1}{|S_n|}\|z^{(n)} - g_{\mathcal{P}}^{(n)}\|^2.$$

We will show that this functional is invariant with respect to a whole-numbered scaling of S_n. In the sequel, we omit the indices (n) and γ every now and then if it is appropriate. Moreover, we identify $g_{\mathcal{P}}^{(n)}$ with its values on the index set S_n and write shortly $g_s^{(n)}$ $(s \in S_n)$ for the value $g_r^{(n)}(s)$ where the partition element $r \in \mathcal{P}$ is chosen such that $s \in r$.

We consider sequences $(\gamma_n)_{n\in\mathbb{N}}$ of parameters and wedgelet segmentations $(\hat{\mathcal{P}}^{(n)}, \hat{g}^{(n)})$ that minimize the Potts functional $H_{\gamma_n}^{(n)}(z^{(n)}, \cdot)$ for $n \in \mathbb{N}$. We ask, whether, depending on the function f and the parameter sequence, the limit of $\hat{g}^{(n)}$ for $n \to \infty$ has something to do with the original signal f, or if there is even some sort of convergence. This will be made more precise in the next paragraph.

5.1.2 Discrete and Continuous Domains

Because, for the analysis of consistency, discrete data has to be compared with continuous results, and vice versa, operators for discretization and embedding are introduced and examined in this Paragraph.

Notation 5.1.1: *Let in the following for each $n \in \mathbb{N}$ the symbols $\boldsymbol{\iota}^n$ and $\boldsymbol{\delta}^n$ denote the following operators:*

$$\boldsymbol{\iota}^n : \mathbb{R}^{S_n} \longrightarrow L^2([0,1)^2),\ z \mapsto \sum_{s\in S_n} z_s \cdot \mathbb{1}_{I_s^{(n)}}, \tag{5.3}$$

$$\boldsymbol{\delta}^n :\ L^2([0,1)^2) \to \mathbb{R}^{S_n},\ g \longmapsto \left(\frac{1}{\lambda(I_s^{(n)})} \int_{I_s^{(n)}} g\, d\lambda \right)_{s\in S_n}. \tag{5.4}$$

The operator $\boldsymbol{\iota}^n$ maps a real valued signal z on the discrete rectangle S_n to a real valued function on the continuous rectangle $[0,1)^2$, while conversely by $\boldsymbol{\delta}^n$ a function $g \in L^2([0,1)^2)$ is transformed to a real valued image on S_n. We will omit the index n in $\boldsymbol{\iota}^n$ and write $\boldsymbol{\iota}$ where convenient.

Let $\mathcal{P}^{(n)}$ be a partition of S_n. It is immediate from the definition, that the set (of sets) $\mathcal{P} = \{\bigcup_{t\in r} \bigcup_{s\in I_t^{(n)}} s : r \in \mathcal{P}^{(n)}\}$ is a partition of $[0,1)^2$. This leads to the following extension of Notation 5.1.1.

Notation 5.1.2: *We will apply $\boldsymbol{\iota}^n$ also to segmentations $(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})$ of S_n with the definitions*

$$\boldsymbol{\iota}^n(\mathcal{P}^{(n)}) = \{\bigcup_{t\in r} \bigcup_{s\in I_t^{(n)}} s : r \in \mathcal{P}^{(n)}\}$$

and

$$\boldsymbol{\iota}^n(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)}) = (\boldsymbol{\iota}^n(\mathcal{P}^{(n)}), \boldsymbol{\iota}^n(g_{\mathcal{P}}^{(n)})).$$

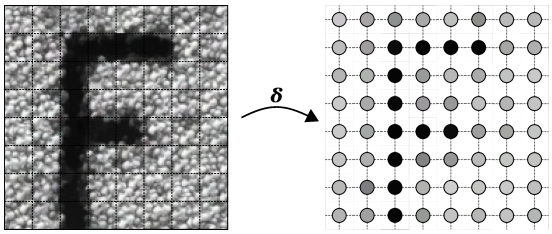


Fig. 5.1: Illustration of the embedding operator $\boldsymbol{\iota}^n$ from Notation 5.1.1

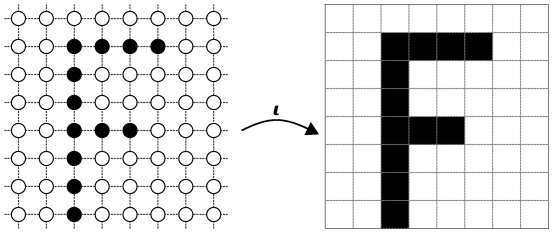


Fig. 5.2: Illustration of the discretization operator $\boldsymbol{\delta}^n$ from Notation 5.1.1

Before we particularize the consistency problem, some results concerning the embedding and discretization are given.

Lemma 5.1.3: *Let $n \in \mathbb{N}$ and $x^{(n)}, y^{(n)} \in \mathbb{R}^{S_n}$. Then*

$$\left\langle \boldsymbol{\iota}^n(x^{(n)}), \boldsymbol{\iota}^n(y^{(n)}) \right\rangle = \frac{1}{|S_n|} \left\langle x^{(n)}, y^{(n)} \right\rangle.$$

Proof.

$$\begin{aligned}\left\langle \boldsymbol{\iota}(x^{(n)}), \boldsymbol{\iota}(y^{(n)}) \right\rangle &= \int (\sum_{s\in S_n} \mathbb{1}_{I_s^{(n)}} x_s^{(n)})(\sum_{t\in S_n} \mathbb{1}_{I_t^{(n)}} y_t^{(n)})\, d\lambda = \sum_{s\in S_n} x_s^{(n)} y_s^{(n)} \lambda(I_s^{(n)}) \\ &= \frac{1}{|S_n|} \sum_{s\in S_n} x_s^{(n)} y_s^{(n)} = \frac{1}{|S_n|} \left\langle x^{(n)}, y^{(n)} \right\rangle\end{aligned}$$

□

An immediate consequence is the subsequent

Corollary 5.1.4: *Let $n \in \mathbb{N}$ and $x^{(n)}, y^{(n)} \in \mathbb{R}^{S_n}$. Then*

$$\|\boldsymbol{\iota}^n(x^{(n)})\|_2^2 = \frac{1}{|S_n|} \|x^{(n)}\|^2.$$

In the following the symbol $\mathfrak{S}$ will stand for a segmentation class over the image domain $[0,1)^2$ and the symbols $\mathfrak{S}^{(n)}$, $1 \leq i \leq n$, will denote segmentation classes over the index sets S_n. For the moment one may have in mind the wedgelet segmentations $\mathfrak{W}$ and $\mathfrak{W}^{(n)}$ with, for example, constant regression over the segments.

Note that in general neither for a segmentation $(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)}) \in \mathfrak{S}^{(n)}$ the embedding $\boldsymbol{\iota}^n(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})$ has to be contained in $\mathfrak{S}$, nor is the discretization $\boldsymbol{\delta}^n(\mathcal{P}, g_{\mathcal{P}})$ of a segmentation $(\mathcal{P}, g_{\mathcal{P}}) \in \mathfrak{S}$ necessarily contained in $\mathfrak{S}^{(n)}$. We repeat and extend the notation of the Potts functional:

Notation 5.1.5: *Let $\gamma > 0$ and $\mathfrak{F}^n = \{\boldsymbol{\iota}^n(g^{(n)}) : (\mathcal{P}^{(n)}, g^{(n)}) \in \mathfrak{S}^{(n)}\}$, $n \in \mathbb{N}$. Let in the sequel the functionals $H_\gamma^{(n)} : \mathbb{R}^{S_n} \times \mathfrak{S}^{(n)} \to \mathbb{R}$, $\bar{H}_\gamma : L^2((0,1]^2) \times \mathfrak{S} \to \mathbb{R}$ and $\tilde{H}^{(n)} : L^2([0,1)^2) \times \mathfrak{S} \to \mathbb{R}$ be defined by*

$$H_\gamma^{(n)}(z, (\mathcal{P}^{(n)}, g_{\mathcal{P}^{(n)}}^{(n)})) = \gamma \cdot |\mathcal{P}^{(n)}| + \frac{1}{|S_n|}\|z - g_{\mathcal{P}^{(n)}}^{(n)}\|_2^2,$$

$$\bar{H}_\gamma(f, (\mathcal{P}, g_{\mathcal{P}})) = \gamma \cdot |\mathcal{P}| + \|f - g_{\mathcal{P}}\|_2^2 \text{ and}$$

$$\tilde{H}_\gamma^{(n)}(f, (\mathcal{P}, g_{\mathcal{P}})) = \begin{cases} \gamma|\mathcal{P}| + \|f - g_{\mathcal{P}}\|_2^2 & \text{if } g_{\mathcal{P}} \in \mathfrak{F}^{(n)}, \\ \infty & \text{otherwise.} \end{cases}$$

In the following the relationship of the functionals H, $\bar{H}$ and $\tilde{H}$ will be investigated. Natural candidates for discrete wedgelet segmentations that potentially minimize $H_\gamma^{(n)}(z, \cdot)$ in the limit $n \to \infty$ are continuous segmentations $(\mathcal{P}, g_{\mathcal{P}}) \in \mathfrak{S}$ (for convenient $\mathfrak{S}$) minimizing the Potts functional $\bar{H}_\gamma(f, (\mathcal{P}, g_{\mathcal{P}}))$. An intermediate case is the embedding of the discrete segmentations into the continuous set $[0,1)^2$ corresponding to the functional $\tilde{H}_\gamma^{(n)}(f, (\mathcal{P}, g_{\mathcal{P}}))$.

The following two results show how the different cases of continuous and discrete domains can be pulled together.

Lemma 5.1.6: *Let $n \in \mathbb{N}$ and $(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)}) \in \mathfrak{S}^{(n)}$. Then for each $z^{(n)} \in \mathbb{R}^{S_n}$ it holds that*

$$H_\gamma^{(n)}(z^{(n)}, (\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})) = \tilde{H}_\gamma^{(n)}(\boldsymbol{\iota}^n(z^{(n)}), \boldsymbol{\iota}^n(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})).$$

Proof. Since $\boldsymbol{\iota}^n(z^{(n)}) - \boldsymbol{\iota}^n(g_{\mathcal{P}}^{(n)}) = \boldsymbol{\iota}^n(z^{(n)} - g_{\mathcal{P}}^{(n)})$, by Corollary 5.1.4 the following holds:

$$\begin{aligned}\tilde{H}_\gamma^{(n)}(\boldsymbol{\iota}(z^{(n)}), \boldsymbol{\iota}(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})) &= |\boldsymbol{\iota}(\mathcal{P}^{(n)})| + \|\boldsymbol{\iota}(z^{(n)}) - \boldsymbol{\iota}(g_{\mathcal{P}}^{(n)})\|_2^2 \\ &= |\mathcal{P}^{(n)}| + \frac{1}{|S_n|}\|z^{(n)} - g_{\mathcal{P}}^{(n)}\|_2^2 = H_\gamma^{(n)}(z^{(n)}, (\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})).\end{aligned}$$

□

Lemma 5.1.6 implies that the functional $H_\gamma^{(n)}$ is invariant with respect to a scaling of the image domain.

Lemma 5.1.7: *Let $n \in \mathbb{N}$, $f \in L^2([0,1)^2)$ and $(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)}) \in \mathfrak{S}^{(n)}$. Then*

$$(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)}) \in \operatorname{argmin} H_\gamma^n(\boldsymbol{\delta}^{(n)}(f), \cdot)$$

if and only if

$$\boldsymbol{\iota}^{(n)}(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)}) \in \operatorname{argmin} \tilde{H}_\gamma^n(f, \cdot).$$

Proof. First,

$$\begin{aligned}\tilde{H}_\gamma^{(n)}(f, \boldsymbol{\iota}(\mathcal{P}^{(n)}, g^{(n)})) &= \gamma|\boldsymbol{\iota}(\mathcal{P}^{(n)})| + \|f - \boldsymbol{\iota}(g^{(n)})\|_2^2 \\ &= \gamma|\mathcal{P}^{(n)}| + \int (f - \sum_{s \in S_n} \mathbb{1}_{I_s^{(n)}} g_s^{(n)})^2 \, d\lambda \\ &= \gamma|\mathcal{P}^{(n)}| + \|f\|_2^2 + \frac{\|g^{(n)}\|_2^2}{|S_n|} - 2\sum_s g_s^{(n)} \int_{I_s^{(n)}} f \, d\lambda.\end{aligned}$$

Second,

$$\begin{aligned}H_\gamma^{(n)}(\boldsymbol{\delta}^{(n)}(f), (\mathcal{P}^{(n)}, g^{(n)})) &= \gamma|\mathcal{P}^{(n)}| + \frac{1}{|S_n|}\|\boldsymbol{\delta}^{(n)}(f) - g^{(n)}\|^2 \\ &= \gamma|\mathcal{P}^{(n)}| + \frac{1}{|S_n|}\sum_{s \in S_n}(|S_n| \int_{I_s^{(n)}} f \, d\lambda - g_s^{(n)})^2 \\ &= \gamma|\mathcal{P}^{(n)}| + |S_n| \cdot \|f\|_2^2 + \frac{\|g^{(n)}\|_2^2}{|S_n|} - 2\sum_s g_s^{(n)} \int_{I_s^{(n)}} f \, d\lambda.\end{aligned}$$

Therefore $\tilde{H}_\gamma^{(n)}(f, \boldsymbol{\iota}^{(n)}(\mathcal{P}^{(n)}, g^{(n)})) - H_\gamma^{(n)}(\boldsymbol{\delta}^{(n)}(f), (\mathcal{P}^{(n)}, g^{(n)})) = (1 - |S_n|)\|f\|_2^2$ is independent of $(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})$ and the proof is complete. □

Lemma 5.1.7 states that the functional $\tilde{H}_\gamma^{(n)}$ is a good candidate for a continuous version of the Potts functional $H_\gamma^{(n)}$. This functional will therefore be used later on to compare the minimizers of $\bar{H}_\gamma$ with the limit of the minimizers of $H_\gamma^{(n)}$.

5.1.3 Subgaussian Noise

In this paragraph the basic requirements on the noise and some consequences are formulated.

Hypothesis 5.1.8: *The triangular array* $(\xi_s^{(n)})_{s\in S_n, n\in\mathbb{N}}$ *of random variables obeys the following properties:*

(B1) *For all* $n \in \mathbb{N}$ *the random variables* $(\xi_s^{(n)})_{s\in S_n}$ *are independent.*

(B2) *There is a universal constant* $\beta > 0$ *such that for all* $n \in \mathbb{N}$ *and each* $s \in S_n$ *the following inequality holds:*

$$\mathbb{E}(e^{\nu\xi_s^{(n)}}) \leq e^{\beta\nu^2} \quad \forall\ \nu \in \mathbb{R}.$$

In the following, random variables fulfilling condition (B2) will be called **subgaussian**. Now some characterizations of subgaussian random variables are presented:

Lemma 5.1.9: *Subgaussian random variables have mean zero.*

Proof. Let X be subgaussian. Then $\mathbb{E}(e^{\nu X}) \leq e^{\beta\nu^2}$ for all $\nu \in \mathbb{R}$. A Taylor series expansion yields

$$1 + \nu\mathbb{E}(X) + \nu^2\mathbb{E}(X^2) + O(\nu^3) \leq 1 + \nu^2\beta + O(\nu^3).$$

Thus there is some $c > 0$ such that for all $\nu > 0$ the inequality

$$\mathbb{E}(X) \leq \nu(\beta - \mathbb{E}(X^2)) + c\nu^2$$

holds which implies that $\mathbb{E}(X) = 0$. □

Lemma 5.1.10: *A gaussian random variable* X *with variance* σ^2 *and mean* $\mu = 0$ *is subgaussian with constant* $\beta = \frac{\sigma^2}{2}$.

Proof.

$$\begin{aligned}\mathbb{E}(e^{\nu X}) &= \frac{1}{\sigma\sqrt{2\pi}} \int e^{\nu x} e^{-\frac{x^2}{2\sigma^2}}\, dx = \frac{1}{\sigma\sqrt{2\pi}} \int e^{-\frac{1}{2\sigma^2}(x^2 - 2\nu\sigma^2 x + \nu^2\sigma^4 - \nu^2\sigma^4)}\, dx \\ &= e^{\frac{\nu^2\sigma^2}{2}} \frac{1}{\sigma\sqrt{2\pi}} \int e^{-\frac{1}{2\sigma^2}(x-\nu\sigma^2)^2}\, dx = e^{\frac{\nu^2\sigma^2}{2}}\end{aligned}$$

□

Lemma 5.1.11: *Let X be a linear combination of $n \in \mathbb{N}$ independent subgaussian random variables X_i with subgaussianity constants β_i, $1 \leq i \leq n$ and let $\lambda_i \in \mathbb{R}$, $1 \leq i \leq n$. Then the random variable $X = \sum_{i=1}^n \lambda_i X_i$ is subgaussian with constant $\beta = \sum_{i=1}^n \lambda_i^2 \beta_i$.*

Proof.

$$\mathbb{E}(e^{\nu X}) = \mathbb{E}(e^{\nu \sum_{i=1}^n \lambda_i X_i}) = \prod_{i=1}^n \mathbb{E}(e^{\nu \lambda_i X_i}) \leq \prod_{i=1}^n e^{\nu^2 \lambda_i^2 \beta_i} = e^{\nu^2 \sum_{i=1}^n \lambda_i^2 \beta_i} = e^{\nu^2 \beta}$$

□

Lemma 5.1.11 in particular implies that if the random variable X is subgaussian, then also its negative, $-X$, is subgaussian. An important property of the proposed noise is the following estimation of the tail probability of subgaussian random variables.

Lemma 5.1.12: *Let $n \in \mathbb{N}$ and a family of real values $\mu_s \in \mathbb{R}$, $s \in S_n$, be given. Then for all $c \in \mathbb{R}$ the following inequality holds:*

$$\mathbb{P}(|\sum_{s \in S_n} \mu_s \xi_s^{(n)}| \geq c) \leq 2e^{-\frac{c^2}{4\beta \sum_s \mu_s^2}}.$$

Proof. Since

$$\mathbb{P}(\sum_{x \in S_n} \mu_s \xi_s^{(n)} \geq c) = \mathbb{P}(e^{\lambda \sum_{s \in S_n} \mu_s \xi_s^{(n)}} \geq e^{\lambda c})$$

holds for all $\lambda > 0$ we can apply the Markov Inequality and obtain

$$\mathbb{P}(\sum_{s \in S_n} \mu_s \xi_s^{(n)} \geq c) \leq \frac{\mathbb{E}(e^{\lambda \sum_s \mu_s \xi_s^{(n)}})}{e^{\lambda c}}.$$

Condition (B2) together with Lemma 5.1.11 yields

$$\mathbb{E}(e^{\lambda \sum_{s \in S_n} \mu_s \xi_s^{(n)}}) \leq e^{\beta \lambda^2 \sum_s \mu_s^2},$$

and therefore

$$\mathbb{P}(\sum_{s \in S_n} \mu_s \xi_s^{(n)} \geq c) \leq e^{\beta \mu_s^2 \lambda^2 - \lambda c}. \tag{5.5}$$

From $\lambda\mu^2\beta - \lambda c = \beta\left((\lambda\mu - \frac{c}{2\beta\mu})^2 - (\frac{c}{2\beta\mu})^2\right)$ it is deduced that $\lambda = \frac{c}{2\beta\mu_s^2}$ gives an optimal estimate. Using this estimate in equation (5.5) yields

$$\mathbb{P}(\sum_{s\in S_n} \mu_s \xi_s^{(n)} \geq c) \leq e^{-\frac{c^2}{4\beta\sum_s \mu_s^2}}.$$

Replacing μ_s by $-\mu_s$ for all $s \in S_n$ yields

$$\mathbb{P}(-\sum_{s\in S_n} \mu_s \xi_s^{(n)} \geq c) = \mathbb{P}(\sum_{s\in S_n} \mu_s \xi_s^{(n)} \leq -c).$$

Together we obtain

$$\mathbb{P}(|\sum_{s\in S_n} \mu_s \xi_s^{(n)}| \geq c) \leq 2 \cdot e^{-\frac{c^2}{4\beta\sum_s \mu_s^2}}. \tag{5.6}$$

This completes the proof. □

5.2 The Main Theorem

In this section consistency is presented from an abstract point of view. This implies that nothing is assumed about the underlying segmentations a priori but particular conditions are given as the case arises. This result provides the basis for the treatment of wedge segmentations in the next section. We derive the theorems by a generalization of the methods developed in Boysen et al. (2004).

5.2.1 Maximal Inequality for Projections of Noise

In this paragraph we explore the performance of the noise depending on $n \in \mathbb{N}$ for segmentations over index sets S_n that are embedded into $L^2([0,1)^2)$. The following observation is the key tool for estimates on wedgelet partitions. Recall that for some subspace $\mathcal{H} \subset \mathbb{R}^k$ ($k \in \mathbb{N}$) the symbol $\mathit{\Pi}_{\mathcal{H}}$ denotes the projection onto $\mathcal{H}$.

Theorem 5.2.1: *Let for each $n \in \mathbb{N}$ the symbol S_n denote an arbitrary finite index set. Assume that the triangular array $\xi^{(n)} = (\xi_s^{(n)})_{s\in S_n}$ of random variables fulfills conditions* (B1) *and* (B2) *on page 129. Consider for each*

$n \in \mathbb{N}$ a set $\mathfrak{H}_n$ of subspaces of $\mathbb{R}^{S_n}$.
If there is a number $D \in \mathbb{N}$ such that $\dim \mathcal{F} \leq D$ for all $\mathcal{F} \in \bigcup_{n\in\mathbb{N}} \mathfrak{H}_n$ and if $\frac{1}{|\mathfrak{H}_n|} = O(n^{-\alpha})$ for some $\alpha > 0$ then

$$\sup_{n\in\mathbb{N}} \sup_{\mathcal{F}\in\mathfrak{H}_n} \frac{\|\Pi_{\mathcal{F}}\xi^{(n)}\|_2^2}{\log|\mathfrak{H}_n|} < \infty \quad a.s. \tag{5.7}$$

Remark 5.2.2: *Theorem 5.2.1 admits the following interpretation with respect to a segmentation class: Assume that each subspace $\mathcal{F} \in \mathfrak{H}_n$ corresponds to a local regression over a subset of S_n using $\dim \mathcal{F}$ parameters. Moreover, assume that these subsets are exactly the fragments of all partitions of S_n. A solution of the local regression problem (3.1) is given by the projection onto the space $\mathcal{F}$ and – say – corresponds to the family of admissible functions over the respective subspace. If the dimensions of the particular local regression models are globally bounded, and if there are at least polynomially (in n) many subspaces, i.e. at least polynomially many fragments, then the inequality (5.7) holds for the noise. This inequality implies that there is some random variable M such that almost surely for all $n \in \mathbb{N}$ and subspaces $\mathcal{F} \in \mathfrak{H}_n$ the inequality*

$$\|\Pi_{\mathcal{F}}\xi^{(n)}\|_2^2 \leq M \cdot \log|\mathfrak{H}_n|$$

holds. Roughly spoken this means: If the segmentation class is rich enough then the norm of the projection of the noise on any segment can be uniformly estimated by the logarithm of the number of fragments.

Proof of Theorem 5.2.1. Fix $n \in \mathbb{N}$ and $\mathcal{F} \in \mathfrak{H}_n$. Let $(e_i)_{1\leq i\leq \dim\mathcal{F}}$ be a basis of $\mathcal{F}$. Observe that

$$\sum_{i=1}^{\dim\mathcal{F}} |\left\langle \xi^{(n)}, e_i \right\rangle|^2 > z^2 \log|\mathfrak{H}_n|$$

implies, for at least one $1 \leq i \leq \dim\mathcal{F}$, the inequality

$$|\left\langle \xi^{(n)}, e_i \right\rangle|^2 > z^2 \log\frac{|\mathfrak{H}_n|}{\dim\mathcal{F}}.$$

We use this (at inequality $(*)$ in the following lines) to derive the inequality

$$\mathbb{P}\left(\|\Pi_{\mathcal{F}}\xi^{(n)}\|^2 > z^2\log|\mathfrak{H}_n|\right) = \mathbb{P}\left(\sum_{i=1}^{\dim\mathcal{F}} |\left\langle \xi^{(n)}, e_i \right\rangle|^2 > z^2\log|\mathfrak{H}_n|\right)$$

$$\overset{(*)}{\leq} \mathbb{P}\left(\bigcup_{i=1}^{\dim \mathcal{F}} \{|\left\langle \xi^{(n)}, e_i \right\rangle|^2 > \frac{z^2 \log|\mathfrak{H}_n|}{\dim \mathcal{F}}\}\right) \leq \sum_{i=1}^{\dim \mathcal{F}} \mathbb{P}\left(|\left\langle \xi^{(n)}, e_i \right\rangle|^2 > \frac{z^2 \log|\mathfrak{H}_n|}{\dim \mathcal{F}}\right)$$
$$= \sum_{i=1}^{\dim \mathcal{F}} \mathbb{P}\left(|\sum_{s\in S_n} \xi_s^{(n)} e_{i,s}| > z\sqrt{\frac{\log|\mathfrak{H}_n|}{\dim \mathcal{F}}}\right) \overset{(**)}{\leq} 2\sum_{i=1}^{\dim \mathcal{F}} \exp \frac{-z^2 \log|\mathfrak{H}_n|}{4\beta \cdot \dim \mathcal{F} \cdot \sum_s e_{i,s}^2}$$
$$= 2 \cdot \dim \mathcal{F} \cdot |\mathfrak{H}_n|^{\frac{-z^2}{4\beta \dim \mathcal{F}}} \leq 2 \cdot D|\mathfrak{H}_n|^{\frac{-z^2}{4\beta \dim \mathcal{F}}},$$

where inequality $(**)$ in the previous formula is due to Lemma 5.1.12. Therefore

$$\sum_{n\in\mathbb{N}} \sum_{\mathcal{F}\in\mathfrak{H}_n} \mathbb{P}(\|\Pi_{\mathcal{F}}\xi^n\|^2 > z^2 \log|\mathfrak{H}_n|) \leq \sum_{n\in\mathbb{N}} \sum_{\mathcal{F}\in\mathfrak{H}_n} 2 \cdot D \cdot |\mathfrak{H}_n|^{\frac{-z^2}{4\beta\cdot\dim\mathcal{F}}}$$
$$\leq 2D \sum_{n\in\mathbb{N}} |\mathfrak{H}_n| \cdot |\mathfrak{H}_n|^{\frac{-z^2}{4\beta D}} = 2D \sum_{n\in\mathbb{N}} |\mathfrak{H}_n|^{1-\frac{z^2}{4\beta D}} \leq 2D \sum_{n\in\mathbb{N}} n^{-\alpha(\frac{z^2}{4\beta D}-1)}.$$

For large enough $z > 0$ that implies

$$\sum_{n\in\mathbb{N}} \sum_{\mathcal{F}\in\mathfrak{H}_n} \mathbb{P}(\|\Pi_{\mathcal{F}}\xi^n\|^2 > z^2 \log|\mathfrak{H}_n|) < \infty$$

and thus the Borel Cantelli Lemma can be applied resulting in

$$\mathbb{P}(\frac{\|\Pi_{\mathcal{F}}\xi^n\|^2}{\log|\mathfrak{H}_n|} > z^2 \text{ infinitely often}) = 0.$$

Therefore

$$\sup_{n\in\mathbb{N}} \sup_{\mathcal{F}\in\mathfrak{H}_n} \frac{\|\Pi_{\mathcal{F}}\xi^n\|_2^2}{\log|\mathfrak{H}_n|} < \infty \text{ a.s.}$$

which completes the proof. □

An immediate consequence of the previous theorem is the following

Corollary 5.2.3: *Let the assumptions of Theorem 5.2.1 be fulfilled. If it holds that*

- $O(|\mathfrak{H}_n|) = O(n^\tau)$ *for some* $\tau > 0$ *and all* $n \in \mathbb{N}$ *then*

$$\sup_{n\in\mathbb{N}} \sup_{\mathcal{F}\in\mathfrak{H}_n} \frac{\|\Pi_{\mathcal{F}}\xi^n\|_2^2}{\log n} < \infty \quad a.s.$$

- $O(|\mathfrak{H}_n|) = O(\tau^n)$ *for some* $\tau > 0$ *and all* $n \in \mathbb{N}$ *then*

$$\sup_{n\in\mathbb{N}} \sup_{\mathcal{F}\in\mathfrak{H}_n} \frac{\|\Pi_{\mathcal{F}}\xi^n\|_2^2}{n} < \infty \quad a.s.$$

In the next paragraph we will provide the link between segmentations and the relatively abstract result of Theorem 5.2.1.

Before we can apply the results to a segmentation class, we need some preparation. Let a partition $\mathcal{P}^{(n)}$ of S_n be given and a family of function spaces $\mathcal{F}_r^{(n)}$, $r \in \mathcal{P}$, such that $\operatorname{supp} f \subset r$ for each $f \in \mathcal{F}_r^{(n)}$ and each $r \in \mathcal{P}$. Then $\mathcal{F}_r^{(n)} \perp \mathcal{F}_{r'}^{(n)}$ for all $r, r' \in \mathcal{P}^{(n)}$ with $r \neq r'$. This observation justifies the following

Notation 5.2.4: *Consider the direct orthogonal sum $\mathcal{F}_{\mathcal{P}^{(n)}}^{(n)} = \bigoplus_{r \in \mathcal{P}^{(n)}} \mathcal{F}_r^{(n)}$. We denote the projection to $\mathcal{F}_{\mathcal{P}^{(n)}}^{(n)}$ in the following shortly by $\Pi_{\mathcal{P}^{(n)}}$.*

Now we can make precise what was suggested in Remark 5.2.2 and apply Theorem 5.2.1 to a segmentation class.

Lemma 5.2.5: *For each $n \in \mathbb{N}$, let a class $\mathfrak{P}^{(n)}$ of partitions of S_n be given and let $\mathcal{R}^{(n)} \supseteq \bigcup_{\mathcal{P} \in \mathfrak{P}^{(n)}} \mathcal{P}$. Consider for each $n \in \mathbb{N}$ and each $r \in \mathcal{R}^{(n)}$ a class of function spaces $\mathcal{F}_r^{(n)} \subset \mathbb{R}^{S_n}$ such that $\operatorname{supp} f \subset r$ for all $f \in \mathcal{F}_r^{(n)}$. If there is a global constant $D \in \mathbb{N}^+$ with $\dim \mathcal{F}_r^{(n)} \leq D$ for all $r \in \mathcal{R}^{(n)}$ and all $n \in \mathbb{N}$, and if there is some $\alpha > 0$ such that $|\mathcal{R}^{(n)}|^{-1} = O(n^{-\alpha})$ for all $n \in \mathbb{N}$, then there is a random variable M such that for all $n \in \mathbb{N}$ and all partitions $\mathcal{P}^{(n)} \in \mathfrak{P}^{(n)}$ of S_n the following inequality holds:*

$$\|\Pi_{\mathcal{P}^{(n)}} \xi^{(n)}\|^2 \leq M \cdot |\mathcal{P}^{(n)}| \cdot \log |\mathcal{R}^{(n)}|. \tag{5.8}$$

Proof. Let $f \in \mathbb{R}^{S_n}$. The identity

$$\Pi_{\mathcal{P}^{(n)}} f = \sum_{r \in \mathcal{P}^{(n)}} \Pi_{\mathcal{F}_r^{(n)}} f$$

implies that

$$\|\Pi_{\mathcal{P}^{(n)}} f\|^2 = \sum_{r \in \mathcal{P}^{(n)}} \|\Pi_{\mathcal{F}_r^{(n)}} f\|^2.$$

We set $\mathfrak{H}_n := \{\mathcal{F}_r^{(n)} : r \in \mathcal{R}^{(n)}\}$ and apply Theorem 5.2.1: There is a random variable M such that for each $r \in \mathcal{P}^{(n)}$:

$$\|\Pi_{\mathcal{F}_r^{(n)}} f\|^2 \leq M \cdot \log |\mathfrak{H}_n| = M \cdot \log |\mathcal{R}^{(n)}|.$$

Therefore

$$\|\Pi_{\mathcal{P}^{(n)}} \xi^{(n)}\|^2 \leq \sum_{r \in \mathcal{P}^{(n)}} M \cdot \log |\mathcal{R}^{(n)}| = M \cdot |\mathcal{P}^{(n)}| \cdot \log |\mathcal{R}^{(n)}|$$

which completes the proof. □

Note that interestingly the inequality (5.8) in Lemma 5.2.5 does not contain the cardinality $|\mathfrak{P}^{(n)}|$ of the set of partitions, but only the number of possible fragments $|\mathcal{R}^{(n)}|$ of partitions.

5.2.2 Projective Segmentation Classes

After the short excursion to the properties of the projection of the noise we come back to (global) estimations for the minimization of the Potts functional. We first need some preparation.

Definition 5.2.6 (Projective Segmentation Class): *Let a finite set S_n, a set $\mathcal{R}^{(n)}$ of subsets of S_n, a partition class $\mathfrak{P}^{(n)}$ with $\mathcal{P}^{(n)} \subset \mathcal{R}^{(n)}$ for each $\mathcal{P}^{(n)} \in \mathfrak{P}^{(n)}$ and a finite-dimensional function space $\mathcal{G} \subset L^2([0,1)^2)$ be given. Let $\delta^{(n)}\mathcal{G} := \{\delta^{(n)} f : f \in \mathcal{G}\}$. Assume that for each $r \subset S_n$ the function space $\mathcal{F}_r^{(n)}$ is given by $\mathcal{F}_r^{(n)} = \{\mathbb{1}_r f : f \in \delta^{(n)}\mathcal{G}\}$. Then the class of segmentations $\mathfrak{S} = (\mathfrak{P}^{(n)}, \mathcal{F}^{(n)})$ will be called a* **projective $\mathcal{F}-$segmentation class** *of S_n with fragments $\mathcal{R}^{(n)}$.*

One difference to the class of segmentations introduced in Chapter 1 is that the functions $f \in \mathcal{F}$ here have full domain S but restricted support. Since a function space $\mathcal{F}' \subset \mathbb{R}^r$ can be identified with a space of functions on $\mathbb{R}^S$ with support on r, it is only a different representation we deal with. The other difference is that the function spaces $\mathcal{F}_r$, $r \subset S$, result from a global function space $\mathcal{G}$ by discretization and restriction.

With the definition

$$\mathcal{P} \wedge \mathcal{Q} := \{p \cap q : p \in \mathcal{P}, q \in \mathcal{Q}\}, \quad \mathcal{P}, \mathcal{Q} \in \mathfrak{P},$$

we get the following structural result:

Lemma 5.2.7: *Let $(\mathfrak{P}, \mathcal{F})$ be a projective segmentation class. Let $\mathcal{P}, \mathcal{Q} \in \mathfrak{P}$. Let $f \in \bigoplus_{r\in\mathcal{P}} \mathcal{F}_r$ and $g \in \bigoplus_{r\in\mathcal{Q}} \mathcal{F}_r$. Then any linear combination of f and g is contained in $\bigoplus_{r\in\mathcal{P}\wedge\mathcal{Q}} \mathcal{F}_r$.*

Proof. Let $\lambda, \mu \in \mathbb{R}$. Since $\mathcal{P}$ and $\mathcal{Q}$ are partitions the following holds:

$$\lambda f + \mu g = \sum_{r\in\mathcal{P}} \lambda f_r + \sum_{q\in\mathcal{Q}} \mu g_q = \sum_{r\in\mathcal{P}} \sum_{\substack{q\in\mathcal{Q}:\\ q\cap r\neq\emptyset}} \lambda \cdot \mathbb{1}_{r\cap q} f_r + \sum_{q\in\mathcal{Q}} \sum_{\substack{r\in\mathcal{P}:\\ q\cap r\neq\emptyset}} \mu \cdot \mathbb{1}_{r\cap q} g_q$$

$$= \sum_{\substack{r\in\mathcal{P},q\in\mathcal{Q}:\\ r\cap q\neq\emptyset}} \mathbb{1}_{r\cap q}(\lambda f_r + \mu g_q)$$

By the definition of a projective segmentation class for each $r \in \mathcal{P}$ and $q \in \mathcal{Q}$ with $r \cap q \neq \emptyset$ it holds that $\mathbb{1}_{q\cap r} f_r \in \mathcal{F}_{r\cap q}$ and $\mathbb{1}_{q\cap r} g_q \in \mathcal{F}_{r\cap q}$. Therefore $\lambda \cdot \mathbb{1}_{q\cap r} f_r + \mu \cdot \mathbb{1}_{q\cap r} g_q \in \mathcal{F}_{r\cap q}$ and thus $\lambda f + \mu g \in \bigoplus_{r\in\mathcal{P}\wedge\mathcal{Q}} \mathcal{F}_r$. □

Lemma 5.2.8: *Let $n \in \mathbb{N}$, a projective segmentation class $\mathfrak{S}_n$ of S_n, a signal $f \in L^\infty([0,1)^2)$ and a vector $\xi^{(n)} \in \mathbb{R}^{S_n}$ be given. Consider*

$$(\hat{\mathcal{P}}^{(n)}, \hat{f}^{(n)}) \in \operatorname*{argmin}_{(\mathcal{P}^{(n)}, f_{\mathcal{P}}^{(n)})\in\mathfrak{S}_n} H_\gamma^{(n)}(\boldsymbol{\delta}^n(f) + \xi^{(n)}, (\mathcal{P}^{(n)}, f_{\mathcal{P}}^{(n)})) \tag{5.9}$$

and let $(\mathcal{Q}^{(n)}, g_{\mathcal{Q}}^{(n)}) \in \mathfrak{S}_n$. Then the following holds:

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq 2\gamma(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + 3\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|_2^2 + 8\frac{\|\Pi_{\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}}\xi^{(n)}\|^2}{|S_n|}. \tag{5.10}$$

Proof. By assumption (5.9) and Lemma 5.1.7 the embedded segmentation $\boldsymbol{\iota}(\hat{\mathcal{P}}^{(n)}, \hat{f}_{\mathcal{P}}^{(n)})$ is a minimizer of $\tilde{H}_\gamma^{(n)}(f + \boldsymbol{\iota}(\xi^{(n)}), \cdot\,)$ and therefore

$$\begin{aligned}
&\tilde{H}_\gamma^{(n)}(f + \boldsymbol{\iota}(\xi^{(n)}), \boldsymbol{\iota}(\mathcal{P}^{(n)}, \hat{f}_{\mathcal{P}}^{(n)})) = \gamma|\mathcal{P}^{(n)}| + \int (\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - (f + \boldsymbol{\iota}(\xi^{(n)})))^2\, d\lambda \\
&\leq \tilde{H}_\gamma^{(n)}(f + \boldsymbol{\iota}(\xi^{(n)}), \boldsymbol{\iota}(\mathcal{P}^{(n)}, g_{\mathcal{P}}^{(n)})) = \gamma|\mathcal{Q}^{(n)}| + \int (\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - (f + \boldsymbol{\iota}(\xi^{(n)})))^2\, d\lambda
\end{aligned}$$

which implies

$$\begin{aligned}
&\gamma|\hat{\mathcal{P}}^{(n)}| + \|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f - \boldsymbol{\iota}(\xi^{(n)})\|_2^2 \leq \gamma|\mathcal{Q}^{(n)}| + \|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f - \boldsymbol{\iota}(\xi^{(n)})\|_2^2 \\
\iff &\gamma|\hat{\mathcal{P}}^{(n)}| + \|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 + 2\left\langle \boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f, \boldsymbol{\iota}(\xi^{(n)})\right\rangle + \|\boldsymbol{\iota}(\xi^{(n)})\|_2^2 \\
&\qquad \leq \gamma|\mathcal{Q}^{(n)}| + \|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|_2^2 + 2\left\langle \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f, \boldsymbol{\iota}(\xi^{(n)})\right\rangle + \|\boldsymbol{\iota}(\xi^{(n)})\|_2^2 \\
\iff &\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \\
&\qquad \leq \gamma(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + \|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|^2 + 2\left\langle \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - \boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}), \boldsymbol{\iota}(\xi^{(n)})\right\rangle.
\end{aligned}$$

Observe that, as $\mathfrak{S}_n$ is a projective segmentation class, Lemma 5.2.7 can be applied and $f^{(n)} - g^{(n)} \in \bigoplus_{r\in\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}} \mathcal{F}_r$ and therefore $\Pi_{\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}}(f^{(n)} - g^{(n)}) = f^{(n)} - g^{(n)}$, thus by Lemma 5.1.3 we obtain

$$\left\langle \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - \boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}), \boldsymbol{\iota}(\xi^{(n)})\right\rangle = \frac{1}{|S_n|}\left\langle \Pi_{\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}}(f^{(n)} - g^{(n)}), \xi^{(n)}\right\rangle$$

$$
\begin{aligned}
&= \frac{1}{|S_n|} \left\langle f^{(n)} - g^{(n)}, \Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)}) \right\rangle \\
&\leq \|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)})\| \cdot |S_n|^{-1/2} \|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\| \\
&\leq \frac{\|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\|}{|S_n|^{1/2}} \cdot \|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\| + \frac{\|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\|}{|S_n|^{1/2}} \|f - \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)})\|.
\end{aligned}
$$

We use the inequality $ab \leq a^2 + (1/4)b^2$, $a, b \in \mathbb{R}$, derived from $(a - b/2)^2 \geq 0$ and get

$$
\begin{aligned}
\left\langle \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - \boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}), \boldsymbol{\iota}(\xi^{(n)}) \right\rangle \leq & \frac{\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2}{4} + \frac{\|f - \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)})\|_2^2}{4} \\
& + 2 \frac{\|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\|^2}{|S_n|}.
\end{aligned}
$$

Putting the two inequalities together yields

$$
\begin{aligned}
\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|^2 &\leq \gamma(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + \|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|^2 + \frac{1}{2}\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|^2 \\
&\quad + \frac{1}{2}\|f - \boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)})\|^2 + 4|S_n|^{-1}\|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\|^2 \\
\frac{1}{2}\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|^2 &\leq \gamma(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + \frac{3}{2}\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|^2 + 4\frac{\|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\|_2^2}{|S_n|} \\
\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|^2 &\leq 2\gamma(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + 3\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|^2 + 8\frac{\|\Pi_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}}(\xi^{(n)})\|_2^2}{|S_n|}.
\end{aligned}
$$

This completes the proof. □

5.2.3 The Main Theorems

We can now formulate the first consistency result:

Theorem 5.2.9: *Let a finite-dimensional function space $\mathcal{G} \subset L^2([0,1)^2)$ and for each $n \in \mathbb{N}$ a finite set S_n and a projective $\mathcal{G}-$segmentation class $\mathfrak{S}_n = (\mathfrak{P}^{(n)}, \mathcal{F}^{(n)})$ with fragments $\mathcal{R}^{(n)}$ over S_n be given. Assume that*

- *$|S_n| \to \infty$ for $n \to \infty$,*
- *there is a real number $\varrho > 0$ such that $|\mathcal{R}^{(n)}|^{-1} = O(n^{-\varrho})$ for all $n \in \mathbb{N}$,*
- *$(\gamma_n)_{n \in \mathbb{N}}$ is a sequence of real numbers with $\gamma_n \to 0$ and $\frac{\gamma_n |S_n|}{\log |\mathcal{R}^{(n)}|} \to \infty$ for $n \to \infty$.*

Let $f \in L^2([0,1)^2)$ *with*

$$\inf_{k\in\mathbb{N}} \limsup_{n\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}(g') - f\|_2^2 = 0$$

and let the triangular array $(\xi^{(n)})_{n\in\mathbb{N}}$ *of random variables obey conditions* (B1) *and* (B2). *Consider*

$$(\hat{\mathcal{P}}^{(n)}, \hat{f}^{(n)}) \in \operatorname*{argmin}_{(\mathcal{P}',f'_{\mathcal{P}})\in\mathfrak{S}_n} H^{(n)}_{\gamma_n}\left(\boldsymbol{\delta}^n(f) + \xi^{(n)}, (\mathcal{P}', f'_{\mathcal{P}})\right)$$

for each $n \in \mathbb{N}$. *Then almost surely*

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}) - f\|_2^2 \longrightarrow 0 \quad \text{for } n \to \infty.$$

Proof. We consider some $(\mathcal{Q}^{(n)}, g^{(n)}_{\mathcal{Q}}) \in \mathfrak{S}_n$. By Lemma 5.2.8

$$\|\boldsymbol{\iota}(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 \le 2\gamma_n(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + 3\|\boldsymbol{\iota}(g^{(n)}_{\mathcal{Q}}) - f\|_2^2 + 8\frac{\|\Pi_{\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}}\xi^{(n)}\|_2^2}{|S_n|}.$$

Let $\mathcal{V}^{(n)} := \{r \cap q : r, q \in \mathcal{R}^{(n)}\}$. Then $|\mathcal{R}^{(n)}| \le |\mathcal{V}^{(n)}| \le |\mathcal{R}^{(n)}|^2$. Therefore $|\mathcal{V}^{(n)}|^{-1} = O(n^{-\varrho})$ and $\log|\mathcal{V}^{(n)}| \le 2\log|\mathcal{R}^{(n)}|$. For two partitions $\mathcal{P}$ and $\mathcal{Q}$ it holds that $|\mathcal{P}\wedge\mathcal{Q}| \le |\mathcal{Q}|\cdot|\mathcal{P}|$. By Lemma 5.2.5, there is a random variable M such that almost surely

$$\begin{aligned}\|\Pi_{\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}}\xi^{(n)}\|_2^2 &\le M\cdot|\hat{\mathcal{P}}^{(n)}\wedge\mathcal{Q}^{(n)}|\cdot\log|\mathcal{V}^{(n)}|\\ &\le M\cdot(|\hat{\mathcal{P}}^{(n)}|\cdot|\mathcal{Q}^{(n)}|)\log|\mathcal{R}^{(n)}|\end{aligned}$$

Putting the two inequalities together with $C' = 8\cdot M$ yields the following:

$$\begin{aligned}\|\boldsymbol{\iota}(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 \le& |\hat{\mathcal{P}}^{(n)}|(-2\gamma_n + C'|\mathcal{Q}^{(n)}|\frac{\log|\mathcal{R}^{(n)}|}{|S_n|}) + 2\gamma_n|\mathcal{Q}^{(n)}|\\ &+ 3\|\boldsymbol{\iota}(g^{(n)}_{\mathcal{Q}}) - f\|_2^2.\end{aligned}$$

Now fix $k \in \mathbb{N}$ and let $\mathcal{Q}^{(n)} = k$ for all $n \in \mathbb{N}$ large enough. Letting $\frac{\gamma_n|S_n|}{\log|\mathcal{R}^{(n)}|} \to \infty$ for $n \to \infty$ implies that $2\gamma_n \ge C'k\frac{\log|\mathcal{R}^{(n)}|}{|S_n|}$ for large enough $n \in \mathbb{N}$. Therefore for large enough $n \in \mathbb{N}$:

$$\|\boldsymbol{\iota}(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 \le 2\gamma_n\cdot k + 3\|\boldsymbol{\iota}(g^{(n)}_{\mathcal{Q}}) - f\|_2^2. \tag{5.11}$$

Because $(\mathcal{Q}^{(n)}, g^{(n)}_{\mathcal{Q}})$ can be chosen freely, we have

$$\limsup_{n\to\infty}\|\boldsymbol{\iota}(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 \le \inf_{k\in\mathbb{N}}\limsup_{n\to\infty}\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\le k}}\left(2\gamma_n\cdot k + 3\|\boldsymbol{\iota}(g^{(n)}_{\mathcal{Q}}) - f\|_2^2\right)$$

$$= \inf_{k\in\mathbb{N}} \limsup_{n\to\infty} 4\gamma_n k + \inf_{k\in\mathbb{N}} \limsup_{n\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\le k}} 3\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|_2^2 = 0$$

This completes the proof. □

Hypothesis 5.2.10: *Consider for each $n \in \mathbb{N}$ a class $\mathfrak{P}^{(n)}$ of partitions. We will use the following condition frequently:*

(C1) *There is some constant $C > 0$ such that for each $n \in \mathbb{N}$ and all $\mathcal{P}^{(n)}, \mathcal{Q}^{(n)} \in \mathfrak{P}^{(n)}$ it holds that*

$$|\mathcal{P}^{(n)} \wedge \mathcal{Q}^{(n)}| \le C(|\mathcal{P}^{(n)}| + |\mathcal{Q}^{(n)}|).$$

Now we state a result for the almost sure rate of convergence.

Theorem 5.2.11: *Let a finite-dimensional function space $\mathcal{G} \subset L^2([0,1)^2)$, and for each $n \in \mathbb{N}$ a finite set S_n and a projective $\mathcal{G}-$segmentation class $\mathfrak{S}_n = (\mathfrak{P}^{(n)}, \mathcal{F}^{(n)})$ with fragments $\mathcal{R}^{(n)}$ over S_n be given. Assume that*

- *$|S_n| \to \infty$ for $n \to \infty$,*
- *$\mathfrak{P}^{(n)}$ fulfills condition* (C1),
- *there is a real number $\varrho > 0$ such that $|\mathcal{R}^{(n)}|^{-1} = O(n^{-\varrho})$ for all $n \in \mathbb{N}$.*

Let $f \in L^2([0,1)^2)$ and assume that there are some real values $\alpha > 0$, $\theta \ge 0$, a real valued function $F : \mathbb{R} \to \mathbb{R}$ with $\lim_{x\to\infty} F(x) = \infty$ and a constant $C > 0$ such that

$$\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}_n:\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}(g') - f\| \le C \cdot \left(\frac{k^\theta}{F(n)} + \frac{1}{k^\alpha}\right) \tag{5.12}$$

for all $n \in \mathbb{N}$. Let the triangular array $(\xi^{(n)})_{n\in\mathbb{N}}$ of random variables obey conditions (B1) *and* (B2). *Assume $(\gamma_n)_{n\in\mathbb{N}}$ is a sequence of real numbers with $\frac{\gamma_n|S_n|}{\log|\mathcal{R}^{(n)}|} \to \infty$ for $n \to \infty$. Consider*

$$(\hat{\mathcal{P}}^{(n)}, \hat{f}^{(n)}) \in \operatorname*{argmin}_{(\mathcal{P}',f'_{\mathcal{P}})\in\mathfrak{S}_n} H_{\gamma_n}^{(n)}\left(\boldsymbol{\delta}^n(f) + \xi^{(n)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

for each $n \in \mathbb{N}$. Then almost surely

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 = O\left(\min\left\{\gamma_n^{\frac{2\alpha}{2\alpha+1}}, \gamma_n F(n)^{\frac{1}{\theta+\alpha}}\right\}\right). \tag{5.13}$$

Remark 5.2.12: *The previous theorem admits the following interpretation: The inequality (5.12) is an estimate for the approximation of the signal f by a segmentation. The second term $1/k^\alpha$ on the right-hand side of the inequality stands for the error of a best approximation of f within a class of continuous segmentations corresponding to the segmentation classes $\mathfrak{S}^{(n)}$, $n \in \mathbb{N}$. The first term $k^\theta/F(n)$ stands for the error that arises from the discretization of the aforementioned best approximation. If the parameter γ_n is not decreasing too fast, then the convergence rate is given by equation (5.13). In equation (5.13) again the interplay between the approximation error and the discretization error can be observed.*

Proof of Theorem 5.2.11. We consider some $(\mathcal{Q}^{(n)}, g_{\mathcal{Q}}^{(n)}) \in \mathfrak{S}_n$. By Lemma 5.2.8

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq 2\gamma_n(|\mathcal{Q}^{(n)}| - |\hat{\mathcal{P}}^{(n)}|) + 3\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|_2^2 + 8\frac{\|\mathit{\Pi}_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}} \xi^{(n)}\|_2^2}{|S_n|}.$$

Let $\mathcal{V}^{(n)} := \{r \cap q : r, q \in \mathcal{R}^{(n)}\}$. Then $|\mathcal{R}^{(n)}| \leq |\mathcal{V}^{(n)}| \leq |\mathcal{R}^{(n)}|^2$. Therefore $|\mathcal{V}^{(n)}|^{-1} = O(n^{-\varrho})$ and $\log|\mathcal{V}^{(n)}| \leq 2\log|\mathcal{R}^{(n)}|$. By Lemma 5.2.5 there is a random variable M and by Hypothesis 5.2.10 there is a constant C such that almost surely

$$\begin{aligned}\|\mathit{\Pi}_{\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}} \xi^{(n)}\|_2^2 &\leq M \cdot |\hat{\mathcal{P}}^{(n)} \wedge \mathcal{Q}^{(n)}| \cdot \log|\mathcal{V}^{(n)}| \\ &\leq 2C \cdot M \cdot (|\hat{\mathcal{P}}^{(n)}| + |\mathcal{Q}^{(n)}|) \log|\mathcal{R}^{(n)}|\end{aligned}$$

Putting the two inequalities together with $C' = 16 \cdot C \cdot M$ yields the following:

$$\begin{aligned}\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq &|\hat{\mathcal{P}}^{(n)}|(-2\gamma_n + C'\frac{\log|\mathcal{R}^{(n)}|}{|S_n|}) + |\mathcal{Q}^{(n)}|(2\gamma_n + C'\frac{\log|\mathcal{R}^{(n)}|}{|S_n|}) \\ &+ 3\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|_2^2.\end{aligned}$$

$\frac{\gamma_n|S_n|}{\log|\mathcal{R}^{(n)}|} \to \infty$ for $n \to \infty$ implies that $2\gamma_n \geq C'\frac{\log|\mathcal{R}^{(n)}|}{|S_n|}$ for large enough $n \in \mathbb{N}$. Therefore for large enough $n \in \mathbb{N}$:

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq 4\gamma_n \cdot |\mathcal{Q}^{(n)}| + 3\|\boldsymbol{\iota}(g_{\mathcal{Q}}^{(n)}) - f\|_2^2. \tag{5.14}$$

By these assumptions, there is a constant C such that for any sequence $(k_n)_{n\in\mathbb{N}}$ the inequality

$$\inf_{\substack{(g',\mathcal{Q})\in\mathfrak{S}_n:\\ |\mathcal{Q}|\leq k_n}} \|\boldsymbol{\iota}(g') - f\| \leq C \cdot \left(\frac{k_n^\theta}{F(n)} + \frac{1}{k_n^\alpha}\right)$$

holds for each $n \in \mathbb{N}$. Because $(a+b)^2 \leq (a+b)^2 + (a-b)^2 = 2a^2 + 2b^2$ this implies

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq 4\gamma_n k_n + 6C\left(\frac{k_n^{2\theta}}{F(n)^2} + \frac{1}{k_n^{2\alpha}}\right).$$

Thus there is a constant C' such that

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq C'\left(\gamma_n k_n + \frac{k_n^{2\theta}}{F(n)^2} + \frac{1}{k_n^{2\alpha}}\right).$$

Setting $\gamma_n k_n = \frac{1}{k_n^{2\alpha}}$ yields $k_n = \gamma_n^{-\frac{1}{2\alpha+1}}$ and

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq C'\left(2\gamma_n^{\frac{2\alpha}{2\alpha+1}} + \frac{\gamma_n^{-\frac{2\theta}{2\alpha+1}}}{F(n)^2}\right).$$

Setting $\frac{k_n^{2\theta}}{F(n)^2} = \frac{1}{k_n^{2\alpha}}$ yields $k_n = (F(n))^{\frac{1}{\theta+\alpha}}$ and

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 \leq C'\left(\gamma_n F(n)^{\frac{1}{\theta+\alpha}} + 2F(n)^{\frac{-2\alpha}{\alpha+\theta}}\right).$$

Using the last two inequalities we obtain

$$\|\boldsymbol{\iota}(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 = O\left(\min\left\{\gamma_n^{\frac{2\alpha}{2\alpha+1}}, \gamma_n F(n)^{\frac{1}{\theta+\alpha}}\right\}\right).$$

The proof is thus complete. □

5.3 Applications

In this section we apply the results of the previous section to the segmentations that have been developed in this thesis. In the first two paragraphs we present relatively abstract results concering consistency and rate of convergence for dyadic wedge segmentations and hierarchic wedge segmentations. These results still postulate certain approximation properties of the segmentation space and the signal. Then we focus on piecewise polynomial approximation and display the corresponding consistency result. In the last subsection we comment on constant approximation and give a prominent example, the so called horizon functions.

5.3.1 Dyadic Wedge Segmentations

Let in this paragraph for each $n \in \mathbb{N}$ the index sets $S_n = \{1, \dots, n\} \times \{1, \dots, n\}$ be given and let for each dyadic $n \in \mathbb{N}$ the symbols $\mathfrak{P}^{(n)}$ and $\mathfrak{W}^{(n)}$ denote the class of dyadic square partitions and dyadic wedge partitions over S_n, respectively.

To apply the Theorems 5.2.9 and 5.2.11 to dyadic wedge partitions we need the following result concerning the superposition $\mathcal{Q} \wedge \mathcal{P}$ of dyadic wedgelet partitions.

Lemma 5.3.1: *For each $d \in \mathbb{N}$, let the set of fragments*

$$\mathcal{R}^{(2^d)} = \bigcup_{\mathcal{P} \in \mathfrak{W}^{(2^d)}} \mathcal{P}$$

be given. Then for each $d \in \mathbb{N}$ the following holds:

$$2^{2d} < |\mathcal{R}^{(2^d)}| < 2^{4d+1}.$$

Proof. There are $4^d = 2^{2d}$ dyadic squares with side length 1. This implies that $2^{2d} < |\mathcal{R}^{(2^d)}|$. Each of the dyadic squares of a partition is at most divided by an intersection with a line. The number of lines in a square with side length 2^d is due to Corollary 3.2.18 bounded from above by 2^{4d}. The number of total possible fragments in $\mathcal{R}^{(2^d)}$ can thus be estimated by

$$\begin{aligned} |\mathcal{R}^{(2^d)}| &\leq \sum_{i=0}^{d} 4^{d-i} \cdot 2^{4i} = 2^{2d} \cdot \sum_{i=0}^{d} 2^{2i} = 2^{2d} \cdot \frac{2^{2(d+1)} - 1}{2^2 - 1} \\ &< \frac{2^{4d+2} - 1}{2} < 2^{4d+1}. \end{aligned}$$

□

An immediate consequence of the previous lemma is:

Corollary 5.3.2: *With the assumptions of Lemma 5.3.1, first, for all $\theta > 0$ it holds that $|\mathcal{R}^{(2^d)}|^{-1} = O(d^{-\theta})$, and second, $\log |\mathcal{R}^{(2^d)}| = O(d)$.*

Proof. For the first statement observe that for all $\theta > 0$ there is some $d \in \mathbb{N}$ such that $2^a \geq a^\theta$ implying $2^{-a} \leq a^{-\theta}$ for all $a \geq d$. □

Lemma 5.3.3: *Let r_1 and r_2 be two dyadic squares, $r_1 \neq r_2$. If $r_1 \cap r_2 \neq \emptyset$ then either $r_1 \subset r_2$ or $r_2 \subset r_1$. In particular, $r_1 \cap r_2$ is a dyadic square.*

Proof. It suffices to prove the statement in one dimension. Let $I_1 = [(i-1)2^{-k_1}, i2^{-k_1})$ and $I_2 = [(j-1)2^{-k_2}, j2^{-k_2})$ $(k_1, k_2 \in \mathbb{N},\ i, j \in \mathbb{N},\ 1 \leq i \leq k_1, 1 \leq j \leq k_2)$. $I_1 \cap I_2 = \emptyset$ is equivalent with

$$(i-1)2^{-k_1} \geq j2^{-k_2} \ \vee \ (j-1)2^{-k_2} \geq i2^{-k_1}.$$

Let without loss of generality, $k_1 \geq k_2$. Then with $l = k_1 - k_2$ the fact $I_1 \cap I_2 \neq \emptyset$ is equivalent with

$$(i-1) < j2^l \ \wedge \ (j-1)2^l < i.$$

Since $i, j \in \mathbb{N}$ and $2^l \geq 1$, also $i \leq j2^l$ and $(j-1)2^l \leq i-1$. Hence

$$(j-1)2^{-k_2} \leq (i-1)2^{-k_1} < i2^{-k_1} \leq j2^{-k_2},$$

that is $I_2 \subset I_1$. And because $k_1 \neq k_2$ then $I_1 \not\subset I_2$. □

The next lemma provides the confirmation of Hypothesis 5.2.10 for dyadic wedge segmentations.

Lemma 5.3.4: *Let $d \in \mathbb{N}$ and $\mathcal{P}, \mathcal{Q} \in \mathfrak{W}^{(2^d)}$. Then*

$$|\mathcal{P} \wedge \mathcal{Q}| \leq 4(|\mathcal{P}| + |\mathcal{Q}|).$$

Proof. By Lemma 5.3.3 for the intersection of two dyadic subsets r_1, r_2 of S_{2^d} it holds that $r_1 \cap r_2 \in \{r_1, r_2, \emptyset\}$. For two dyadic partitions $\mathcal{D}_1$ and $\mathcal{D}_2$ we thus obtain the inequality $|\mathcal{D}_1 \cap \mathcal{D}_2| \leq |\mathcal{D}_1| + |\mathcal{D}_2|$. Two dyadic wedge partitions therefore consist of less or equal $|\mathcal{P}| + |\mathcal{Q}|$ dyadic squares that can each be intersected by a maximum of two lines yielding a maximum of 4 fragments per dyadic square. Altogether we obtain $|\mathcal{P} \wedge \mathcal{Q}| \leq 4(|\mathcal{P}| + |\mathcal{Q}|)$. □

Theorem 5.3.5: *Let a finite-dimensional function space $\mathcal{G} \subset L^2([0,1)^2)$ and for each $d \in \mathbb{N}$ the finite set $S_{2^d} = \{1, \ldots, 2^d\}^2$ and a projective dyadic wedge $\mathcal{G}-$segmentation class $\mathfrak{S}^{(d)} = (\mathfrak{P}^{(d)}, \mathcal{F}^{(d)})$ with fragments $\mathcal{R}^{(d)}$ over S_{2^d} be given. Let $(\gamma_d)_{d \in \mathbb{N}}$ be a sequence of real numbers with $\gamma_d \to 0$ and*

$\frac{\gamma_d 4^d}{d} \to \infty$ *for* $d \to \infty$. *Let* $f \in L^2([0,1)^2)$ *and let the triangular array* $(\xi^{(n)})_{n\in\mathbb{N}}$ *of random variables obey conditions* (B1) *and* (B2). *Consider*

$$(\hat{\mathcal{P}}^{(d)}, \hat{f}^{(d)}) \in \operatorname*{argmin}_{(\mathcal{P}', f'_{\mathcal{P}})\in\mathfrak{S}^{(d)}} H_{\gamma_d}^{(2^d)}\left(\boldsymbol{\delta}^{2^d}(f) + \xi^{(2^d)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

Then the following two statements are true:

(a) If the relation

$$\inf_{k\in\mathbb{N}} \limsup_{d\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(d)},\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}^{2^d}(g') - f\|_2^2 = 0$$

holds then almost surely

$$\|\boldsymbol{\iota}^{2^d}(\hat{f}^{(d)}) - f\|_2^2 \longrightarrow 0 \quad \text{for } d \to \infty.$$

(b) If there are real values $\alpha > 0$, $\theta \ge 0$, *a real valued function* $F : \mathbb{R} \to \mathbb{R}$ *with* $\lim_{x\to\infty} F(x) = \infty$ *and a constant* $C > 0$ *such that*

$$\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(d)}:\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}^{2^d}(g') - f\| \le C \cdot \left(\frac{k^\theta}{F(d)} + \frac{1}{k^\alpha}\right)$$

for all $d \in \mathbb{N}$, *then almost surely*

$$\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(d)}) - f\|_2^2 = O\left(\min\left\{\gamma_d^{\frac{2\alpha}{2\alpha+1}}, \gamma_d F(d)^{\frac{1}{\theta+\alpha}}\right\}\right).$$

Proof. The theorem relies on Theorems 5.2.9 and 5.2.11. We check the remaining assumptions. First, $S_{2^d} \to \infty$ for $d \to \infty$. Second, condition (C1) of Hypothesis 5.2.10 is fulfilled by Lemma 5.3.4. And third, $|\mathcal{R}^{(d)}|^{-1} = O(d^{-\theta})$ for all $\theta > 0$ and $\gamma_d |S_{2^d}|/|\mathcal{R}^{(d)}| \to \infty$ if $\gamma_d 4^d/d \to \infty$ by Corollary 5.3.2. □

In this paragraph we considered wedgelet partitions with local projections on each of the wedgelets. We did not consider quad-tree partitions and treat the wedges as attributes of the local approximation over the dyadic squares. The reason is that then the local approximation would not have been a projection to a linear function space.

5.3.2 Hierarchic Wedge Segmentations

Now we repeat the steps done in the previous paragraph for the hierarchic wedge partitions. Let in this paragraph for each $n \in \mathbb{N}$ the index sets

$S_n = \{1, \dots, n\} \times \{1, \dots, n\}$ be given, and let for each $n \in \mathbb{N}$ the symbols $\mathfrak{P}^{(n)}$ and $\mathfrak{W}^{(n)}$ denote the class of hierarchic square partitions and hierarchic wedge partitions over S_n, respectively. Furthermore, we define the following restriction of the hierarchic square and wedge partitions: For $n \in \mathbb{N}$ and $1 \leq k \leq \infty$ by $\mathfrak{P}^{(n)}_{\leq k}$ we denote the set of hierarchic partitions $\mathcal{P}$ that are of the form $\mathcal{P} = \{r_{ij}, 1 \leq i \leq r, 1 \leq j \leq m_j\}$ as defined in (2.1) where additionally $m_j \leq k$ for all $1 \leq j \leq k$. $\mathfrak{P}^{(n)}_{\leq k}$ is thus the set of partitions where the number of vertical divisions is bounded by k. By $\mathfrak{W}^{(n)}_{\leq k}$ we denote the hierarchic wedge partitions that are developed from the hierarchic wedge partitions $\mathfrak{P}^{(n)}_{\leq k}$.

To apply the Theorems 5.2.9 and 5.2.11 to dyadic wedge partitions we need the following result concerning the superposition $\mathcal{Q} \wedge \mathcal{P}$ of dyadic wedgelet partitions.

Lemma 5.3.6: *Let $2 < a \leq \infty$ and for each $n \in \mathbb{N}$ the set of fragments*

$$\mathcal{R}^{(n)} = \bigcup_{\mathcal{P} \in \mathfrak{W}^{(n)}_{\leq a}} \mathcal{P}$$

be given. Then for each $n \in \mathbb{N}$ the following holds:

$$n^4 < |\mathcal{R}^{(n)}| < (n+2)^6.$$

Proof. There are $\sum_{k=1}^{n} n-k+1 = n^2/2+n/2$ intervals of the form $\{k, \dots, k+l\}$, $0 \leq l \leq n-k$, $1 \leq k \leq n$. Because the number of vertical intervals allowed in a hierarchic partition is greater or equal 3, any interval can be formed in horizontal and vertical directions. Therefore $\mathcal{R}^{(n)} > n^4$. The number of lines in a rectangle with side lengths l_1 and l_2 is, due to Corollary 3.2.18, bounded from above by $l_1 \cdot l_2$. We use this to estimate the number of total wedges with the following formula:

$$\begin{aligned} |\mathcal{R}^{(n)}_{\leq k}| &\leq \sum_{k=1}^{n} \sum_{l_1=1}^{n-k+1} \sum_{m=1}^{n} \sum_{l_2=1}^{n-m+1} l_1 \cdot l_2 = (\sum_{k=1}^{n} \sum_{l=1}^{n-k+1} l)^2 \\ &= (\sum_{k=1}^{n} \frac{(n-k+1)(n-k+2)}{2})^2 < (n(n+1)(n+2))^2 < (n+2)^6. \end{aligned}$$

□

An immediate consequence of the previous lemma is:

Corollary 5.3.7: *With the assumptions of Lemma 5.3.6, first it holds that* $|\mathcal{R}^{(n)}|^{-1} = O(n^{-4})$*, and second* $\log|\mathcal{R}^{(n)}| = O(\log n)$*.*

The next lemma provides the confirmation of Hypothesis 5.2.10 for hierarchic wedge segmentations with bounded vertical division.

Lemma 5.3.8: *Let* $a \in \mathbb{N}$*,* $n \in \mathbb{N}$ *and* $\mathcal{P}, \mathcal{Q} \in \mathfrak{W}^{(n)}_{\leq a}$*. Then*

$$|\mathcal{P} \wedge \mathcal{Q}| \leq 2a(|\mathcal{P}| + |\mathcal{Q}|)$$

Proof. The fragments of a partition of $\{1, \ldots, n\}$ are intervals. Such a partition with m fragments is thus determined by $m-1$ numbers. Let $\mathcal{I}_1, \mathcal{I}_2$ be two such partitions with m_1 and m_2 fragments. Then $|\mathcal{I}_1 \wedge \mathcal{I}_2| \leq (m_1 - 1) + (m_2 - 1) + 1 < m_1 + m_2 = |\mathcal{I}_1| + |\mathcal{I}_2|$. Now let $\mathcal{I}_1$ and $\mathcal{I}_2$ be the horizontal partitions corresponding to $\mathcal{P}$ and $\mathcal{Q}$. Then

$$|\mathcal{P} \wedge \mathcal{Q}| \leq 2a(|\mathcal{I}_1| + |\mathcal{I}_2|) \leq 2a(|\mathcal{P}| + |\mathcal{Q}|),$$

which completes the proof. □

Theorem 5.3.9: *Let a finite-dimensional function space* $\mathcal{G} \subset L^2([0,1)^2)$*, and for each* $n \in \mathbb{N}$ *the finite set* $S_n = \{1, \ldots, n\}^2$ *and a projective hierarchic wedge* $\mathcal{G}-$*segmentation class* $\mathfrak{S}^{(n)} = (\mathfrak{P}^{(n)}_{\leq a}, \mathcal{F}^{(n)})$*,* $2 < a \leq \infty$*, with fragments* $\mathcal{R}^{(n)}$ *over* S_n *be given. Let* $(\gamma_n)_{n\in\mathbb{N}}$ *be a sequence of real numbers with* $\gamma_n \to 0$ *and* $\frac{\gamma_n n^2}{\log n} \to \infty$ *for* $n \to \infty$*. Let* $f \in L^2([0,1)^2)$ *and let the triangular array* $(\xi^{(n)})_{n\in\mathbb{N}}$ *of random variables obey conditions* (B1) *and* (B2)*. Consider*

$$(\hat{\mathcal{P}}^{(n)}, \hat{f}^{(n)}) \in \operatorname*{argmin}_{(\mathcal{P}', f'_{\mathcal{P}}) \in \mathfrak{S}^{(n)}} H^{(n)}_{\gamma_n}\left(\boldsymbol{\delta}^n(f) + \xi^{(n)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

Then the following statements are true:

(a) If the relation

$$\inf_{k\in\mathbb{N}} \limsup_{n\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\leq k}} \|\boldsymbol{\iota}^n(g') - f\|_2^2 = 0$$

holds then almost surely

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}) - f\|_2^2 \longrightarrow 0 \quad \text{for } n \to \infty.$$

(b) If $a < \infty$ and if there are real values $\alpha > 0$, $\theta \geq 0$ and if, furthermore, there are a real valued function $F : \mathbb{R} \to \mathbb{R}$ with $\lim_{x\to\infty} F(x) = \infty$ and a constant $C > 0$ such that

$$\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)}:\\ |\mathcal{Q}|\leq k}} \|\boldsymbol{\iota}^n(g') - f\| \leq C \cdot \left(\frac{k^\theta}{F(n)} + \frac{1}{k^\alpha}\right)$$

for all $n \in \mathbb{N}$, then almost surely

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 = O\left(\min\left\{\gamma_n^{\frac{2\alpha}{2\alpha+1}}, \gamma_n F(n)^{\frac{1}{\theta+\alpha}}\right\}\right).$$

Proof. The theorem relies on Theorems 5.2.9 and 5.2.11. We check the remaining assumptions. First, $S_n \to \infty$ for $n \to \infty$. Second, condition (C1) of Hypothesis 5.2.10 is fulfilled by Lemma 5.3.8. And third, $|\mathcal{R}^{(n)}|^{-1} = O(n^{-4})$ and $\gamma_n|S_n|/|\mathcal{R}^{(n)}| \to \infty$ if $\gamma_n n^2/\log n \to \infty$ by Corollary 5.3.7. □

5.3.3 Piecewise Polynomial Approximations

Let $\mathcal{R}$ and $\mathcal{R}^{(n)}$ be the set of rectangles in $S = [0,1)^2$ and in $S_n = \{1,\dots,n\}^2$, $n \in \mathbb{N}$, respectively. In this paragraph we consider segmentations of $[0,1)^2$ and of S_n with a regression on the classes of admissible functions $\mathcal{F} = (\mathcal{F}_p)_{p\in\mathcal{R}}$ and $\mathcal{F}^{(n)} = (\mathcal{F}_p^{(n)})_{p\in\mathcal{R}^{(n)}}$, where $\mathcal{F}_p \subset \mathbb{R}^p$, $p \in \mathcal{R}$, and $\mathcal{F}_p^{(n)} \subset \mathbb{R}^p$, $p \in \mathcal{R}^{(n)}$, are spaces of polynomial functions of order $m-1$, $m \in \mathbb{N}_+$.

Let $\mathfrak{P}_k$ and $\mathfrak{P}_k^{(n)}$ be the classes of either dyadic or hierarchic square partitions of $[0,1)^2$ and S_n with maximal k elements. In the following the symbols $\mathfrak{S}_k$ and $\mathfrak{S}_k^{(n)}$ denote the space of segmentations with partitions $\mathfrak{P}_k$ and $\mathfrak{P}_k^{(n)}$ and regression over the function spaces $\mathcal{F}$ and $\mathcal{F}^{(n)}$.

Recall that C^m is the space of m times continuously differentiable functions.

Lemma 5.3.10: *Let f be C^m. Then there are constants c, c' such that*

$$\inf\{\|f - \tilde{f}\|_2 : (\mathcal{P},\tilde{f}) \in \mathfrak{S}_k\} \leq ck^{-m/2}$$

and

$$\inf\{\|f - \boldsymbol{\iota}^{(n)}(\tilde{f})\|_2 : (\mathcal{P},\tilde{f}) \in \mathfrak{S}_k^{(n)}\} \leq c'(k^{-m/2} + \frac{1}{n})$$

Proof. Let d be such that $4^d \leq k < 4^{d+1}$. Then a best approximation with 4^d pieces is worse or as good as a best approximation with k pieces. We show the result in uniform norm which is stronger than the L^2-norm.

Divide the square into 4^d subsquares each of length 2^{-d}. This yields a partition that is hierarchical and dyadic at the same time. On each subsquare, approximate f by its Taylor polynomial of order $m-1$. Let for each $s \in \{((r_1 - 1/2)/2^d, (r_2 - 1/2)/2^d), 1 \leq r_1, r_2 \leq 2^d\}$ the symbol $T_s f$ denote the Taylor approximation of f around s. This gives an approximation error of

$$\begin{aligned}\|f - T_s f\|_\infty &\leq \frac{1}{m!}(\sqrt{2} \cdot 2^{-d-1})^m \sup_{x \in [0,1)^2, l \leq m} \|f^{(m)}(x)\| = \mathrm{O}(2^{-d \cdot m}) \\ &\leq \mathrm{O}(k^{-m/2}).\end{aligned}$$

for all s.

For the second part, we estimate the difference between the approximating Taylor polynomial and its discretized version. We apply now the above argument for piecewise constant approximation on a finer $n \times n$-grid to each summand of the Taylor polynomial. Denoting by $\mathcal{P}_m$ the collection of all monomials of order less than m we derive a uniform bound of

$$\begin{aligned}\|\boldsymbol{\iota}^n(\boldsymbol{\delta}^n(T_s f)) - T_s f\|_\infty &\leq \sum_{p \in \mathcal{P}_m} (\sqrt{2}\frac{1}{n}) \sup_{x \in [0,1)^2} \|p^{(1)}(x)\| \sup_{x \in [0,1)^2, l \leq m} \|f^{(l)}(x)\| \\ &= \mathrm{O}(n^{-1})\end{aligned}$$

□

Corollary 5.3.11: *Let $f \in L^2([0,1)^2)$. Then*

$$\inf_{k \in \mathbb{N}} \limsup_{n \to \infty} \inf_{\substack{(\mathcal{Q}, g') \in \mathfrak{S}^{(n)}, \\ |\mathcal{Q}| \leq k}} \|\boldsymbol{\iota}^n(g') - f\|_2^2 = 0$$

Proof. For all $f \in L^2$ and for all $\varepsilon > 0$ there is a function $f_0 \in C^1$ such that $\|f_0 - f\|_2 \leq \varepsilon$. Therefore by Lemma 5.3.10 there is some constant c such that for all k and all $\varepsilon > 0$ the following holds

$$\begin{aligned}\inf_{\substack{(\mathcal{Q}, g') \in \mathfrak{S}^{(n)}, \\ |\mathcal{Q}| \leq k}} \|\boldsymbol{\iota}^n(g') - f\|_2 &\leq \varepsilon + \inf_{\substack{(\mathcal{Q}, g') \in \mathfrak{S}^{(n)}, \\ |\mathcal{Q}| \leq k}} \|\boldsymbol{\iota}^n(g') - f_0\|_2 \\ &\leq \varepsilon + ck^{-1/2} + \frac{1}{n}.\end{aligned}$$

This yields

$$\inf_{k\in\mathbb{N}} \limsup_{n\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\leq k}} \|\boldsymbol{\iota}^n(g') - f\|_2^2 = \inf_{k\in\mathbb{N}} ck^{-1/2} = 0.$$

□

Now we formulate the results regarding consistency for dyadic and hierarchic wedge segmentations separately. We start with the hierarchic segmentations.

Theorem 5.3.12: *Let the function space $\mathcal{G} \subset L^2([0,1)^2)$ consist of polynomials and for each $n \in \mathbb{N}$ let the finite set $S_n = \{1,\dots,n\}^2$ and a projective hierarchic wedge $\mathcal{G}-$segmentation class $\mathfrak{S}^{(n)} = (\mathfrak{P}^{(n)}, \mathcal{F}^{(n)})$, $2 < a \leq \infty$ with fragments $\mathcal{R}^{(n)}$ over S_n be given. Let $(\gamma_n)_{n\in\mathbb{N}}$ be a sequence of real numbers with $\gamma_n \to 0$ and $\frac{\gamma_n n^2}{\log n} \to \infty$ for $n \to \infty$. Let $f \in L^2([0,1)^2)$ and let the triangular array $(\xi^{(n)})_{n\in\mathbb{N}}$ of random variables obey conditions* (B1) *and* (B2)*. Consider*

$$(\hat{\mathcal{P}}^{(n)}, \hat{f}^{(n)}) \in \operatorname*{argmin}_{(\mathcal{P}',f'_{\mathcal{P}})\in\mathfrak{S}^{(n)}} H^{(n)}_{\gamma_n}\left(\boldsymbol{\delta}^n(f) + \xi^{(n)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

Then almost surely

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}) - f\|_2^2 \longrightarrow 0 \quad \text{for } n \to \infty.$$

Proof. The statement is based on the first part of Theorem 5.3.9. By Corollary 5.3.11 it holds that

$$\inf_{k\in\mathbb{N}} \limsup_{n\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\leq k}} \|\boldsymbol{\iota}^n(g') - f\|_2^2 = \inf_{k\in\mathbb{N}} ck^{-m/2} = 0.$$

This is the only condition missing in comparison with Theorem 5.3.9 and the proof is thus complete. □

For dyadic segmentations even a statement about the rate can be included.

Theorem 5.3.13: *Let the function space $\mathcal{G} \subset L^2([0,1)^2)$ consist of polynomials of degree $m \in \mathbb{N}$ and let for each $d \in \mathbb{N}$ the finite set $S_{2^d} = \{1,\dots,2^d\}^2$ and a projective dyadic wedge $\mathcal{G}-$segmentation class $\mathfrak{S}^{(d)} = (\mathfrak{P}^{(d)}, \mathcal{F}^{(d)})$ with fragments $\mathcal{R}^{(d)}$ over S_{2^d} be given. Let $(\gamma_d)_{d\in\mathbb{N}}$ be a sequence of real numbers*

with $\gamma_d \to 0$ *and* $\frac{\gamma_d 4^d}{d} \to \infty$ *for* $d \to \infty$. *Let* $f \in L^2([0,1)^2)$ *and let the triangular array* $(\xi^{(n)})_{n\in\mathbb{N}}$ *of random variables obey conditions* (B1) *and* (B2). *Consider*

$$(\hat{\mathcal{P}}^{(d)}, \hat{f}^{(d)}) \in \operatorname*{argmin}_{(\mathcal{P}', f'_{\mathcal{P}})\in\mathfrak{S}^{(d)}} H_{\gamma_d}^{(2^d)}\left(\boldsymbol{\delta}^{2^d}(f) + \xi^{(2^d)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

Then almost surely

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}) - f\|_2^2 \longrightarrow 0 \quad \text{for } n \to \infty.$$

If additionally $f \in C^{m'}$, $m' \in \mathbb{N}$, *then*

$$\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(d)}) - f\|_2^2 = O(\gamma_d^{\frac{\min\{m+1,m'\}}{\min\{m+2,m'+1\}}}).$$

In particular $\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(d)}) - f\|_2^2 = O\left((d^2/4^d)^{\frac{\min\{m+1,m'\}}{\min\{m+2,m'+1\}}}\right)$.

Proof. The statement is based on the first part of Theorem 5.3.5. By Corollary 5.3.11 it holds that

$$\inf_{k\in\mathbb{N}} \limsup_{n\to\infty} \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}^n(g') - f\|_2^2 = \inf_{k\in\mathbb{N}} ck^{-m/2} = 0.$$

This is the only condition missing in comparison with Theorem 5.3.9 and the first part of the statement is proven.

By Lemma 5.3.10 there is some constant $c > 0$ such that for all k the following holds

$$\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}^n(g') - f\|_2 \le ck^{-(\min\{m,m'-1\}+1)/2} + \frac{1}{n}.$$

Therefore we can set $\theta = 0$, $\alpha = \min\{m, m'-1\}/2$ and $F(d) = 4^d$ in Theorem 5.3.5 yielding

$$\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(d)}) - f\|_2^2 = O\left(\min\left\{\gamma_d^{\frac{2\alpha}{2\alpha+1}}, \gamma_d F(d)^{\frac{1}{\theta+\alpha}}\right\}\right) = O(\gamma_d^{\frac{\min\{m,m'-1\}+1}{\min\{m,m'-1\}+2}}).$$

Setting $\gamma_d = d^2/4^d$ yields the last statement which completes the proof. □

Remark 5.3.14: *The rate given in Theorem 5.3.13 is very close at the theoretical best rate of convergence. The best rate is given as* $\|\boldsymbol{\iota}^n(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 = O(n^{-(m'/(m'+1))})$, *see Stone (1982), compared with our rate* $\|\boldsymbol{\iota}^n(\hat{f}_{\mathcal{P}}^{(n)}) - f\|_2^2 = O((n/log^2 n)^{-(m'/(m'+1))})$ *for dyadic* n *and* $m \ge m' - 1$.

5.3.4 Piecewise Constant Approximations

For piecewise polynomial segmentations we could show consistency and even a rate of convergence for the dyadic wedge model. In this section we will focus on piecewise constant functions and present some results for the prominent horizon functions. We will treat the discretization and the approximation separately in this paragraph.

Consider the space of constant functions $\mathcal{G} = \{f \in L^2([0,1)^2) : f(x) = \mu \ \forall \ x \in [0,1)^2, \mu \in \mathbb{R}\}$. In this paragraph we consider projective $\mathcal{G}$-segmentation classes.

The Discretization Errors

We start with a result that is crucial for the estimation of the discretization error of wedge segmentations. In the sequel for two sets A and B we use the notation $A \ominus B = (A \setminus B) \cup (B \setminus A)$.

Lemma 5.3.15: *Consider $f \in L^\infty([0,1)^2)$ and for each subset $r \subset [0,1)^2$ let the space $\mathcal{F}_r := \{\mathbb{1}_r g : g \in \mathcal{G}\}$ of constant functions be given. Then for two sets $r, r' \in \mathcal{B}([0,1)^2)$ the following holds:*

$$\|\Pi_{\mathcal{F}_r} f - \Pi_{\mathcal{F}_{r'}} f\|_2^2 \leq 2\lambda(r \ominus r') \sup_{x \in [0,1)^2} |f(x)|^2.$$

Proof. Let $\mu_C := \int_C f \, d\lambda$ for all $C \in \mathcal{B}([0,1)^2$. With this notation we obtain

$$\begin{aligned}
&\|\Pi_{\mathcal{F}_r} f - \Pi_{\mathcal{F}_{r'}} f\| = \int \left(\mathbb{1}_r \int_r f \, d\lambda - \mathbb{1}_{r'} \int_{r'} f \, d\lambda \right)^2 d\lambda \\
&= \int_{r \cap r'} (\mu_r - \mu_{r'})^2 \, d\lambda + \int_{r \setminus r'} \mu_r^2 \, d\lambda + \int_{r' \setminus r} \mu_{r'}^2 \, d\lambda \\
&= \lambda(r \cap r') \mu_{r \ominus r'}^2 + \lambda(r \setminus r') \mu_r^2 + \lambda(r' \setminus r) \mu_{r'}^2 \\
&\leq \left(\lambda(r \cap r') \lambda(r \ominus r')^2 + \lambda(r \setminus r') \lambda(r)^2 + \lambda(r' \setminus r) \lambda(r')^2 \right) \sup_{x \in [0,1)^2} |f(x)|^2 \\
&\leq \left(\lambda(r \cap r') \lambda(r \ominus r')^2 + \lambda(r \ominus r') \lambda(r \cup r')^2 \right) \sup_{x \in [0,1)^2} |f(x)|^2 \\
&\leq 2\lambda(r \ominus r') \sup_{x \in [0,1)^2} |f(x)|^2.
\end{aligned}$$

The proof is complete. □

We will use the previous theorem to estimate the discretization error of segmentations. Therefore we introduce the discretization and the discrete boundary of some subset of $[0,1)^2$. Recall for each $n \in \mathbb{N}$ the set $S_n = \{1,\dots,n\}^2$ and for each $s \in S_n$ the sets $I_s^{(n)} := [(s_1-1)/n, s_1/n) \times [(s_2-1)/n, s_2/n)$. The discretization of a set $r \subset [0,1)^2$ will in the following be defined using

$$r^{(n)} = \left\{ s \in S_n : \left(\frac{s_1 - \frac{1}{2}}{n}, \frac{s_2 - \frac{1}{2}}{n} \right) \in r \right\}$$

by

$$\bar{r}^{(n)} = \bigcup_{s \in r^{(n)}} I_s^{(n)}.$$

The discrete boundary will be denoted by

$$\mathrm{bd}^{(n)}(r) := \{ s \in r^{(n)} : \emptyset \neq I_s^{(n)} \cap r \neq I_s^{(n)} \}.$$

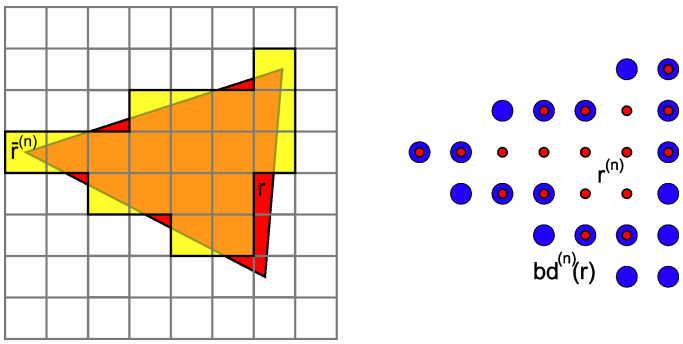


Fig. 5.3: Set r and discretization $\bar{r}^{(n)}$ (left), $r^{(n)}$ and discrete boundary $\mathrm{bd}^{(n)}(r)$ (right).

Corollary 5.3.16: *Let $f \in L^\infty([0,1)^2)$ and let for each subset $r \subset [0,1)^2$ the space $\mathcal{F}_r := \{\mathbb{1}_r g : g \in \mathcal{G}\}$ of constant functions be given. Then there is a constant $c > 0$ such that for all $n \in \mathbb{N}$ and each subset $r \in \mathcal{B}([0,1)^2)$ the inequality*

$$\| \Pi_{\mathcal{F}_r} f - \Pi_{\mathcal{F}_{\bar{r}^{(n)}}} f \|_2^2 \leq c \frac{|\mathrm{bd}^{(n)}(r)|}{|S_n|}$$

holds.

Proof. By the previous lemma we only have to show that $\lambda(r \ominus \bar{r}^{(n)}) \leq |\mathrm{bd}^{(n)}(r)|/|S_n|$. We define $u = \{s \in S_n : I_n^{(n)} \cap r \neq \emptyset\}$ and $l = \{s \in S_n :$

$I_s^{(n)} \subset r\}$. Then $l \subseteq r^{(n)} \subseteq u$ and therefore $\bigcup_{s\in l} I_s^{(n)} \subseteq \bar{r}^{(n)} \subseteq \bigcup_{s\in u} I_s^{(n)}$. Moreover, $\bigcup_{s\in l} I_s^{(n)} \subseteq r \subseteq \bigcup_{s\in u} I_s^{(n)}$ and $\mathrm{bd}^{(n)}(r) = u \setminus l$. This yields

$$r \ominus \bar{r}^{(n)} = (r \setminus \bar{r}^{(n)}) \cup (\bar{r}^{(n)} \setminus r) \subset \bigcup_{s\in u} I_s^{(n)} \setminus \bigcup_{s\in l} I_s^{(n)} = \bigcup_{s\in \mathrm{bd}^{(n)}(r)} I_s^{(n)}.$$

Since $\lambda(I_s^{(n)}) = \frac{1}{|S_n|}$ for each $s \in S_n$ we obtain

$$\lambda(r \ominus \bar{r}^{(n)}) \leq \sum_{s\in bd^{(n)}(r)} \lambda(I_s^{(n)}) = \frac{|\mathrm{bd}^{(n)}(r)|}{|S_n|}.$$

□

Now we comment on a projective $\mathcal{G}$-segmentation class with partitions consisting of rectangles and line-divided rectangles. Recall the notation $\mathit{\Pi}_{\mathcal{F}_\mathcal{P}} f = \sum_{r\in\mathcal{P}} \mathit{\Pi}_{\mathcal{F}_r} f$ and that $S_n = \{1, \ldots, n\}^2$. We will use the following result for the hierarchic segmentation classes.

Lemma 5.3.17: *Let $\mathcal{G}$ be the space of constant functions on $[0,1)^2$. Let for each $r \subset [0,1)^2$ the class of functions $\mathcal{F}_r = \{\mathbb{1}_r g : g \in \mathcal{G}\}$ be given. Let $\mathcal{P}$ be a partition of $[0,1)^2$ with cardinality $k = |\mathcal{P}|$, which consists of rectangles and wedges. Let for each $n \in \mathbb{N}$ the discretized version of $\mathcal{P}$ be given by $\mathcal{P}^{(n)} = \{\bar{r}^{(n)} : r \in \mathcal{P}\}$. Then*

$$\|\mathit{\Pi}_{\mathcal{F}_\mathcal{P}} f - \mathit{\Pi}_{\mathcal{F}_{\mathcal{P}^{(n)}}} f\|_2^2 = O\left(\frac{k}{n}\right).$$

Proof. Since a line in a square with side length $0 < l \leq 1$ can intersect at most $(n \cdot l + 2)$ plates of the form $I_s^{(n)}$, $s \in S_n$, in horizontal and the same number in vertical direction, its discrete length can be estimated by $2n \cdot l + 4$. Since the wedges (and rectangles) have at most 5 edges, the length of the longest possible discrete boundary of $r \in \mathcal{P}$ can thus be estimated by $\max\{\mathrm{bd}^{(n)}(r) : r \in \mathcal{P}\} \leq 10n + 20$. By Corollary 5.3.16 there is constant $c > 0$ such that

$$\begin{aligned}\|\mathit{\Pi}_{\mathcal{F}_\mathcal{P}} f - \mathit{\Pi}_{\mathcal{F}_{\mathcal{P}^{(n)}}} f\|_2^2 = \|\sum_{r\in\mathcal{P}} \mathit{\Pi}_{\mathcal{F}_r} f - \mathit{\Pi}_{\mathcal{F}_{\bar{r}^{(n)}}} f\|_2^2 &\leq \sum_{r\in\mathcal{P}} \|\mathit{\Pi}_{\mathcal{F}_r} f - \mathit{\Pi}_{\mathcal{F}_{\bar{r}^{(n)}}} f\|_2^2 \\ &\leq |\mathcal{P}| c \cdot 10n/n^2 + O(1/n^2).\end{aligned}$$

Thus the discretization error is given as $O(k/n)$. □

For dyadic wedge partitions the discretization error can be estimated even with a better rate.

Lemma 5.3.18: *Let $\mathcal{G}$ be the space of constant functions on $[0,1)^2$. Let for each $r \subset [0,1)^2$ the class of functions $\mathcal{F}_r = \{\mathbb{1}_r g : g \in \mathcal{G}\}$ be given. Let $\mathcal{P}$ be a dyadic wedge partition of $[0,1)^2$ with cardinality $k = |\mathcal{P}|$. Let for each $d \in \mathbb{N}$ the discretized version of $\mathcal{P}$ be given by $\mathcal{P}^{(2^d)} = \{\bar{r}^{(2^d)} : r \in \mathcal{P}\}$. Then*

$$\|\Pi_{\mathcal{F}_\mathcal{P}} f - \Pi_{\mathcal{F}_{\mathcal{P}^{(2^d)}}} f\|_2^2 = O\left(\frac{\sqrt{k}}{2^d}\right).$$

Proof. Recall the definition of the depth of a dyadic partition $\mathcal{P}$ on page 44. Firstly we estimate the length of the discrete boundary of a partition $\mathcal{P}^{(2^d)}$ with depth d. Each rectangle of the form $[(i-1)2^{-l}, i2^{-l}) \times [(j-1)2^{-l}, j2^{-l})$, $1 \le i, j \le d$, has by Lemma 5.3.3 an empty discrete boundary $\mathrm{bd}^{(2^d)}(r)$. Therefore the only boundaries that have to be counted are given by the lines intersecting such rectangles. Let l be the side length of a rectangle. The length of the boundary of a wedge dividing the rectangle can be estimated by $\mathrm{bd}^{(2^d)}(r) \le 2 \cdot l \cdot 2^d$. Let $k = |\mathcal{P}^{(d)}|$ and let $k' \in \mathbb{N}$ such that $4^{k'-1} < k \le 4^{k'}$. We estimate the length $L_\mathcal{P}^{(2^d)}$ by enumerating the fragments as dyadic squares ordered by size and count each dyadic square as two wedges:

$$L_\mathcal{P}^{(2^d)} = \sum_{r \in \mathcal{P}} \mathrm{bd}^{(2^d)}(r) \le 2 \sum_{m=0}^{k'} 4^m \cdot 2 \cdot 2^{-m} \cdot 2^d \le 16 \cdot 2^d \cdot 2^{k'-1} < 16 \cdot 2^d \cdot \sqrt{k}.$$

Thus by Corollary 5.3.16 there is a constant $c > 0$ such that

$$\begin{aligned}\|\Pi_{\mathcal{F}_\mathcal{P}} f - \Pi_{\mathcal{F}_{\mathcal{P}^{(n)}}} f\|_2^2 = \|\sum_{r \in \mathcal{P}} \Pi_{\mathcal{F}_r} f - \Pi_{\mathcal{F}_{\bar{r}^{(n)}}} f\|_2^2 &\le \sum_{r \in \mathcal{P}} \|\Pi_{\mathcal{F}_r} f - \Pi_{\mathcal{F}_{\bar{r}^{(n)}}} f\|_2^2 \\ &\le c 2^d \sqrt{k} \frac{1}{4^d} = c \cdot \frac{\sqrt{k}}{2^d}.\end{aligned}$$

Therefore the discretization error is given as $O(\sqrt{k}/2^d)$. □

Horizon Functions

Now we explore (continuous) wedgelet segmentations with respect to suitability for approximation of real valued functions over $[0,1)^2$. For a fundamental analysis of the approximation quality of these segmentations we

consider functions that can be easily specified and characterized: so called **horizon functions**, two-dimensional functions arising from segmentation of $[0,1)^2$ by a one-dimensional function (horizon). We will exploit that one dimensional projections $\tilde{P}$ of the quad tree partition P can be used to estimate the approximation quality of a wedgelet partition over P by the quality of a piecewise (over fragments of $\tilde{P}$) affine approximation of the one dimensional horizon. Before we can state the main result in more detail we have to devise the needed background.

We consider horizon functions and start with their definition.

Definition 5.3.19: *A* **horizon** *is a one dimensional function* $F : [0,1) \to [0,1]$. *The two dimensional associated function* f *defined by*

$$f : [0,1) \times [0,1) \to \mathbb{R}, \quad f(x,y) = \begin{cases} 1 & \text{if } y \leq F(x) \\ 0 & \text{otherwise} \end{cases}$$

is called (two dimensional) **horizon function**.

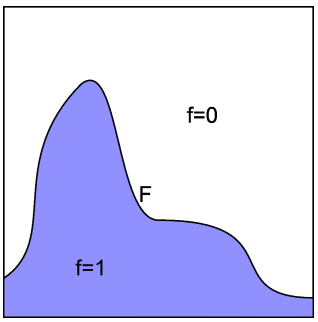


Fig. 5.4: Horizon F with associated horizon function h_F

Lemma 5.3.20: *Consider two horizon functions* $f, g : [0,1)^2 \to \{0,1\}$ *associated to horizons* $F, G : [0,1) \to [0,1)$. *Then*

$$\|F - G\|_1 = \|f - g\|_2^2.$$

Proof. Using Fubinis Theorem we obtain:

$$\begin{aligned} \|f - g\|_2^2 &= \int_{[0,1)^2} (f(u) - g(u))^2 \, d\lambda^2(u) \\ &= \int_{[0,1)} \lambda(\{x : f(x,y) \neq g(x,y)\}) \, dx \end{aligned}$$

$$= \int_{[0,1)} |F(x) - G(x)|\, dx = \|F - G\|_1$$

and the proof is complete. □

Hierarchic Segmentations of Horizon Functions

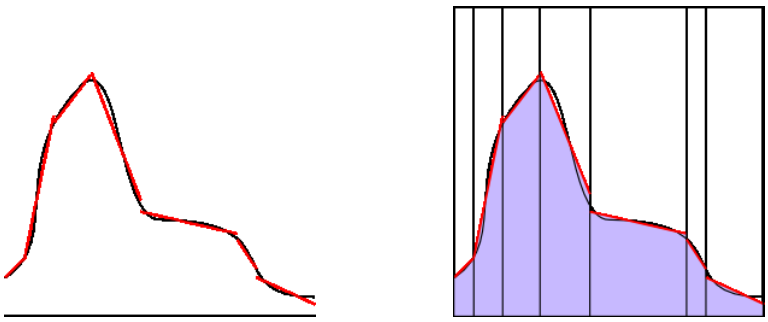


Fig. 5.5: Piecewise affine approximation of horizon F and corresponding hierarchic partition.

Lemma 5.3.21: *Let for each $k \in \mathbb{N}$ the symbol $\mathcal{A}_k$ denote the class of piecewise affine functions from $[0,1)$ to $[0,1)$ with less or equal k pieces. Consider $\alpha > 0$ and let f be a horizon function with horizon F such that*

$$\inf_{G \in \mathcal{A}_k} \|F - G\| = O(k^{-\alpha}).$$

Let $\mathcal{G}$ be a function class that contains the constant functions over $[0,1)^2$. Consider the the projective hierarchic wedge $\mathcal{G}$-segmentation class $\mathfrak{S} = (\mathfrak{P}_{\leq 2}, \mathcal{F})$. Then

$$\inf_{\substack{(\mathcal{P}, g_\mathcal{P}) \in \mathfrak{S}: \\ |\mathcal{P}| \leq k}} \|f - g_\mathcal{P}\|_2^2 = O(k^{-\alpha}).$$

Proof. For each piecewise affine function $G : [0,1) \to [0,1)$ with k pieces there is a partition $\mathcal{P}'$ of $[0,1)$ such that $G|_r : r \to [0,1)$ is an affine function for each $r \in \mathcal{P}'$. Thus for each r the two sets $\{(x,y) \in r \times [0,1) : y \leq G(x)\}$ and $\{(x,y) \in r \times [0,1) : y > G(x)\}$ form a wedge division of the rectangle $r \times [0,1)$. Since the constant functions are contained in $\mathcal{G}$ there is a hierarchic wedge segmentation $(\mathcal{P}, g_\mathcal{P}) \in \mathfrak{S}$ with $|\mathcal{P}| \leq 2k$ and $g_\mathcal{P} = h_G$. Therefore by Lemma 5.3.20

$$\inf_{\substack{(\mathcal{P}, g_\mathcal{P}) \in \mathfrak{S}: \\ |\mathcal{P}| \leq k}} \|h_F - g_\mathcal{P}\|_2^2 \leq \inf_{G \in \mathcal{A}_{k/2}} \|F - G\| = O((k/2)^{-\alpha}) = O(k^{-\alpha}).$$

□

Now we can present the statement concerning the rate convergence for piecewise constant hierarchic segmentations of horizon functions.

Theorem 5.3.22: *Let the function space $\mathcal{G} \subset L^2([0,1)^2)$ consist of constant functions and let for each $n \in \mathbb{N}$ the finite set $S_n = \{1,\dots,n\}^2$ and a projective hierarchic wedge $\mathcal{G}-$segmentation class $\mathfrak{S}^{(n)} = (\mathfrak{P}^{(n)}, \mathcal{F}^{(n)})$, $2 < a \leq \infty$, with fragments $\mathcal{R}^{(n)}$ over S_n be given. Let $(\gamma_n)_{n\in\mathbb{N}}$ be a sequence of real numbers with $\gamma_n \to 0$ and $\frac{\gamma_n n^2}{\log n} \to \infty$ for $n \to \infty$. Let $\alpha > 0$ and f be a two dimensional horizon function with horizon F with*

$$\inf\{\|F - H\|_1 : H \text{ piecewise affine with } k \text{ pieces}\} = O(k^{-\alpha})$$

Let the triangular array $(\xi^{(n)})_{n\in\mathbb{N}}$ of random variables obey conditions (B1) *and* (B2)*. Consider*

$$(\hat{\mathcal{P}}^{(n)}, \hat{f}^{(n)}) \in \operatorname*{argmin}_{(\mathcal{P}',f'_{\mathcal{P}})\in\mathfrak{S}^{(n)}} H^{(n)}_{\gamma_n}\left(\boldsymbol{\delta}^n(f) + \xi^{(n)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

Then almost surely

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 = O\left(\gamma_n^{\frac{2\alpha}{2\alpha+1}}\right).$$

In particular $\|\boldsymbol{\iota}^n(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 = O(\log^2 n/n^2)^{\frac{2\alpha}{2\alpha+1}}$.

Proof. The statement is based on Theorem 5.3.9. By Lemma 5.3.17 and Lemma 5.3.21 it holds that

$$\begin{aligned}\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\leq k}} \|\boldsymbol{\iota}^n(g') - f\|_2 &\leq \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(n)},\\ |\mathcal{Q}|\leq k}} \inf_{\substack{(\mathcal{P},f_{\mathcal{P}})\in\mathfrak{S},\\ |\mathcal{P}|\leq k}} \|\boldsymbol{\iota}^n(g') - f_{\mathcal{P}}\|_2^2 + \|f_{\mathcal{P}} - f\|_2^2 \\ &\leq O(k/n) + O(k^{-\alpha}).\end{aligned}$$

Therefore we can set $\theta = 1$ and $F(n) = n$ in Theorem 5.3.9 yielding

$$\|\boldsymbol{\iota}^n(\hat{f}^{(n)}_{\mathcal{P}}) - f\|_2^2 = O\left(\min\left\{\gamma_n^{\frac{2\alpha}{2\alpha+1}}, \gamma_n \cdot n^{\frac{1}{1+\alpha}}\right\}\right).$$

Setting $\gamma_n = (\log^2 n)/n^2$ yields the particular statement which completes the proof. □

Dyadic Segmentations of Horizon Functions

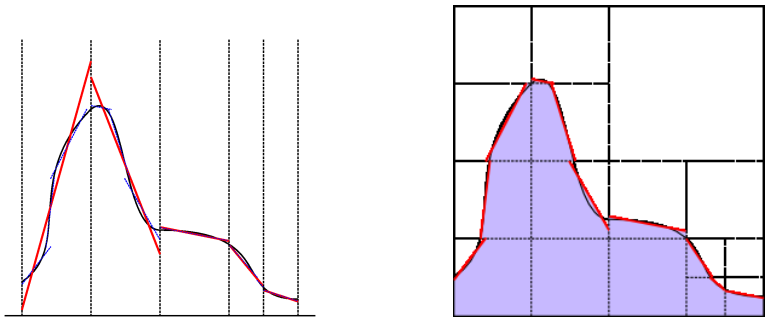


Fig. 5.6: Affine approximation of F over a dyadic partition $\tilde{P}$ and a dyadic wedgelet segmentation.

Theorem 5.3.23: *Let the function space $\mathcal{G} \subset L^2([0,1)^2)$ consist of constant functions and let for each $d \in \mathbb{N}$ the finite set $S_d = \{1,\dots,2^d\}^2$ and a projective dyadic wedge $\mathcal{G}-$segmentation class $\mathfrak{S}^{(d)} = (\mathfrak{P}^{(d)}, \mathcal{F}^{(d)})$, with fragments $\mathcal{R}^{(d)}$ over S_{2^d} be given. Let $\mathfrak{S}$ be the corresponding projective wedge $\mathcal{G}-$segmentation class over $[0,1)^2$. Let $(\gamma_d)_{d\in\mathbb{N}}$ be a sequence of real numbers with $\gamma_d \to 0$ and $\frac{\gamma_d 4^d}{d} \to \infty$ for $d \to \infty$. Let $\alpha > 0$ and $f \in L^2([0,1)^2)$ with*

$$\inf\{\|f-g\|_2^2 : (\mathcal{P},g) \in \mathfrak{S} : |\mathcal{P}| \le k\} = O(k^{-\alpha}). \tag{5.15}$$

Let the triangular array $(\xi^{(n)})_{n\in\mathbb{N}}$ of random variables obey conditions (B1) *and* (B2)*. Consider*

$$(\hat{\mathcal{P}}^{(d)}, \hat{f}^{(d)}) \in \operatorname*{argmin}_{(\mathcal{P}',f'_{\mathcal{P}})\in\mathfrak{S}^{(d)}} H_{\gamma_d}^{(2^d)}\left(\boldsymbol{\delta}^{2^d}(f) + \xi^{(2^d)}, (\mathcal{P}', f'_{\mathcal{P}})\right).$$

Then almost surely

$$\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(d)}) - f\|_2^2 = O\left(\gamma_d^{\frac{2\alpha}{2\alpha+1}}\right).$$

In particular, $\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(d)}) - f\|_2^2 = O(d^2/4^d)^{\frac{2\alpha}{2\alpha+1}}$.

Proof. The statement is based on Theorem 5.3.5. By Lemma 5.3.18 and the assumptions it holds that

$$\begin{aligned}\inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(d)},\\ |\mathcal{Q}|\le k}} \|\boldsymbol{\iota}^{2^d}(g') - f\|_2 &\le \inf_{\substack{(\mathcal{Q},g')\in\mathfrak{S}^{(d)},\\ |\mathcal{Q}|\le k}} \inf_{\substack{(\mathcal{P},f_{\mathcal{P}})\in\mathfrak{S},\\ |\mathcal{P}|\le k}} \|\boldsymbol{\iota}^{2^d}(g') - f_{\mathcal{P}}\|_2^2 + \|f_{\mathcal{P}} - f\|_2^2 \\ &\le O(\sqrt{k}/2^d) + O(k^{-\alpha}).\end{aligned}$$

Therefore we can set $\theta = 1/2$ and $F(d) = 2^d$ in Theorem 5.3.5 yielding

$$\|\boldsymbol{\iota}^{2^d}(\hat{f}_{\mathcal{P}}^{(2^d)}) - f\|_2^2 = O\left(\min\left\{\gamma_d^{\frac{2\alpha}{2\alpha+1}}, \gamma_d \cdot 2^{d\cdot\frac{1}{\alpha+1/2}}\right\}\right).$$

Setting $\gamma_d = d^2/4^d$ yields the particular statement which completes the proof. □

Remark 5.3.24: *The assumption (5.15) in the previous theorem has been validated for a class of horizon functions induced by certain Hölder functions by Donoho, see Donoho (1999), p. 871. In particular, if for a function F in C^1 there are a constant c and a real value $1 < \alpha \leq 2$ such that*

$$|\frac{d}{dx}F(x) - \frac{d}{dx}F(y)| \leq c \cdot |x-y|^{\alpha-1},\ x, y \in [0,1)^2, \tag{5.16}$$

then assumption (5.15) holds for the corresponding horizon function f.

We expect that there is a wider class of functions leading to the same or a better result for the following reasons. Firstly, Donoho uses an equidistant dyadic one-dimensional grid for the approximation of a function $[0,1) \to [0,1)$ by piecewise affine functions and argues by 'back projection' to a corresponding dyadic wedge segmentation. It is known that the adaptivity of dyadic segmentations would yield a better result. Secondly – and this is rather a limitation of the consideration of horizons – functions and their inverse are treated differently although the two dimensional horizon functions are invariant up to reflection. For example the function $x \mapsto x^2$ is a function with $\alpha = 2$ in (5.16) above but $x \mapsto x^{\frac{1}{2}}$ is not at all contained in the class.

The result for the hierarchic segmentations is better than that for the dyadic segmentations because the considered class of approximated functions is much richer. This had to be expected since the hierarchic segmentations exactly provide affine horizons. Nevertheless, the computational effort for this model makes it less usable in practice, compare the runtime measurements in the next chapter.

5.4 Synopsis

In this chapter we have explored wedge segmentations resulting from minimizing Potts functionals with respect to consistency and rates of convergence. We have started with the set up borrowed from nonparametric regression. Then we have devised the needed background concerning discretization and embedding of data and segmentations and have specified the basic requirement on the noise. One crucial ingredient for the consistency and rates of convergence is the maximal inequality (5.7) for the projection of the noise. This result is valid under very broad conditions and it implies that if a segmentation class is rich enough then the projection of the noise on any segment can be uniformly estimated by the logarithm of the number of segments. On the other side, also estimates of the approximation speed of the underlying signal f by the used projective segmentation class are indispensable. These ingredients are basic to the estimate provided by Lemma 5.2.8. Balancing these two terms leads directly to the main theorems. Firstly, Theorem 5.2.9 provides consistency of the wedge segmentations. Here it was assumed that the signal f may be perfectly approximated for the signal size tending to infinity. Secondly, Theorem 5.2.11 included a rate of convergence. For this result we had to assume that the segmentation class was given such that the cardinality of a superposition of two partitions $\mathcal{P}$ and $\mathcal{Q}$ was given by less than a constant factor of $|\mathcal{P}| + |\mathcal{Q}|$. Additionally, this theorem contains explicit separate assumptions concerning the discretization error and the approximation rate of the segmentation.

These results have been applied to the two crucial segmentation classes of this thesis, the dyadic and hierarchic wedge segmentations. The result has been formulated separately for these two classes. For both classes the number of fragments of a segmentation had to be estimated. For the hierarchic model the superposition of two segmentations is not subadditive in the aforementioned sense. Therefore a result concerning the convergence rate can only be formulated with the restriction that hierarchic partitions have a globally bounded number of vertical divisions of each horizontal stripe.

For piecewise polynomial approximations of a signal $f \in L^2([0,1)^2)$ consistency has been displayed in Theorems 5.3.12 and 5.3.13. A result concerning the convergence rate could only be formulated for the hierarchic model in cause of the restriction just mentioned. Moreover, this result is formulated for dyadic segmentations under the assumption that f is sufficiently smooth, i.e. $f \in C^m$. There a rate near to the theoretical optimal rate could be

presented.

For piecewise constant approximation of a signal $f \in L^2([0,1)^2)$ we treated the discretization error separately from the projection properties. For a special class of functions – the horizon functions – there are results providing a convergence rate. The result for the hierarchic model is insofar better as the class of horizon functions could be chosen much larger than it has been done for the dyadic model in the literature. However, the result concerning complexity, Theorem 4.3.1, and runtime measurements in the next Chapter (pages 169 -174) show that the hierarchic model is of use in real applications only with certain modifications.

6. Experimental Results

In this chapter we will illustrate the theoretical results and algorithms of the previous chapters by way of simulation results.

6.1 Implementation

The algorithms presented in the Chapters 1- 4 have all been implemented. A software package is available together with a graphical user interface. We will firstly comment on the framework the software has been developed in and display some details of its structure. Then we will present results of a runtime analysis.

6.1.1 Platform

The algorithms for the computation of wedge segmentations have been implemented in Oberon. Oberon is the name for both an operating system and a programming language in the Pascal/Modula tradition. The Oberon project was launched in 1985 by N. Wirth and J. Gutknecht, see Wirth and Gutknecht (1992), Gutknecht (1994) and `http://www.oberon.ethz.ch`. The operating system Oberon is a single user, multi-tasking system that runs on bare hardware or on top of a host operating system like Windows, MacOS and Linux.

The software uses functionalities of the packages Voyager and ANTS$_{\text{InFields}}$. Voyager is a project to explore the feasibility of a portable and extensible system for simulation and data analysis systems. The Voyager project is carried out by StatLab Heidelberg and was launched in 1993 by G. Sawitzki, M. Diller, F. Friedrich et al, see Sawitzki (1996). ANTS$_{\text{InFields}}$ is a software package for simulation and statistical inference on Gibbs Fields. It has been developed since 1997 by F. Friedrich, see Friedrich (2002), Friedrich (2003) and `http://www.antsinfields.de`. Although the main focus of ANTS$_{\text{InFields}}$ is on stochastic simulation of Gibbs fields, the basic tools for handling and for the visualization of one and more dimensional data supplied with ANTS$_{\text{InFields}}$ have proved to be very helpful for the implementation of the wedge segmen-

tations.

By use of the framework Oberon the software is portable between different operating systems. The sources of all components, from the system core up to the high level mathematics routines, are contained and accessible. Therefore, the software provides insight into its internal structure at any level. Currently there are versions of the software for Linux, Windows, Bluebottle and NativeOberon provided. They will be made available online on `http://www.antsinfields.de`.

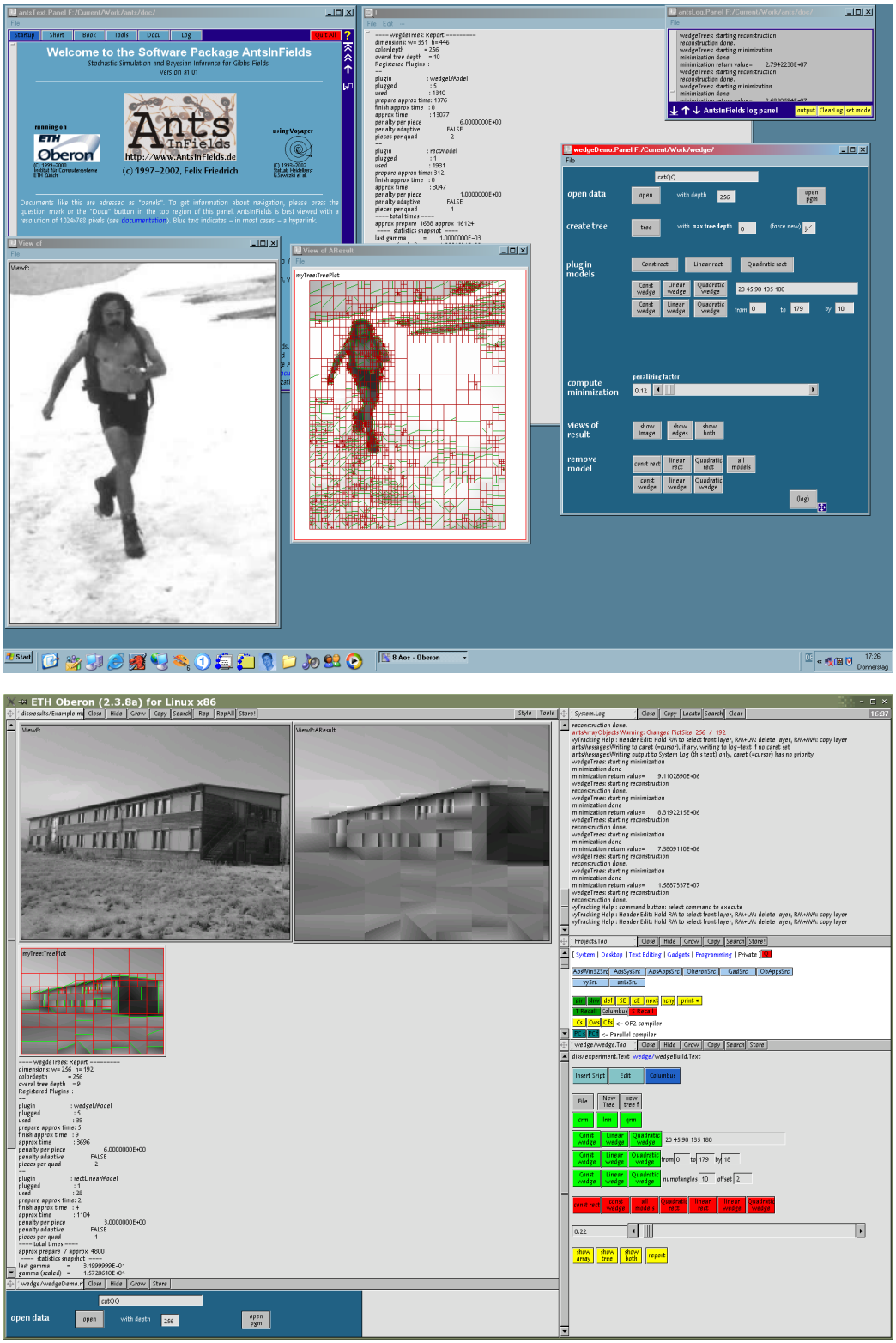


Fig. 6.1: Wedge segmentation with Oberon/Voyager/Ants on Windows (top) and Linux(bottom).

6.1.2 Details

Parts of the algorithms, in particular those concerning the constant models, have been implemented by K. Wicker, see Wicker (2004). The linear regression plug-in has been written by L. Demaret and an extension to quadratic models has been coded by P. Poongpiyapaiboon.

We have separated the implementation of the local regression from the (global) minimization part of the algorithms. In order to achieve this, we provided object definitions with a well defined interface for both, the local regression part and the structures corresponding to the hierarchic partitions and the quad-tree structure, see Figures 6.2 and 6.3.

```
LocalApproximation = OBJECT
   penalty, distance: LONGREAL;
   model: Model;
END;


Model = OBJECT ( . . . )
   penalty: LONGREAL;
   . . .
   PROCEDURE Init; (* constructor *)
   PROCEDURE PrepareApproximation (data: antsArrays.X2d);
   PROCEDURE Approximate (x, y, w, h: LONGINT; VAR approx: LocalApproximation);
   PROCEDURE FinishApproximation;
   PROCEDURE Reconstruct (x, y, w, h: LONGINT; approx: LocalApproximation; reco: antsArrays.X2d);
   . . .
END;
```

Fig. 6.2: Interface of the local regression model. The object LocalApproximation acts as the container for the results of the local regression object.

```
Segmentation = POINTER TO RECORD ( ... )
   PROCEDURE Init; (* constructor *)
   PROCEDURE Create (data: antsArrayObjects.tLONGREALObject; maxdepth: LONGINT);
   PROCEDURE PlugIn (model: Model; name: ARRAY OF CHAR; maxdepth: LONGINT);
   PROCEDURE Minimize (gamma: LONGREAL): LONGREAL;
   PROCEDURE Reconstruct (VAR reco: antsArrays.X2d);
   PROCEDURE Remove (model: Model);
   PROCEDURE RemoveAll;
   PROCEDURE Stats (): StatisticObject;
   ...
END;
```

Fig. 6.3: Interface of the segmentation object. Regression models can be 'plugged' into the segmentation object.

We have made use of the local approximation objects in the modules for the hierarchic model and the quad tree approach with a 'plug-in' mechanism. This has many advantages. Firstly, debugging and testing the code could be done in little separable parts. We tried to follow a component based approach as described in Szyperski (1998). Secondly, new regression models can be programmed and inserted at any time (even at runtime). Equivalently, new kinds of partitions can be implemented and one may make use of the present local approximation models. Thirdly, the strongly object oriented implementation led to a good modeling of the mathematical structure. We could therefore provide a very fast implementation of the dyadic wedge model by applying several models with different angles one after the other. Compare the time measurements in the next section and the comparison with BeamLab. We could even improve the efficiency of the approach of K. Wicker with a speed up factor of about 3, see Wicker (2004), pp. 90-91 and the time measurements below.

Currently provided are modules for regression on rectangles and wedge divided rectangles with an arbitrary set of angles. For constant, linear and quadratic regression, optimized code is implemented. Additionally, there is a module that can do a generic regression on rectangles and wedges as discussed in Subsection 3.1.3. The set of angles can be used in an adaptive way where it is chosen dependent on the depth of the corresponding rectangle. This provides even faster algorithms, compare the time measurements in Subsection 6.2.2 below.

The hierarchic partitions and the dyadic partitions are provided as objects where the local regression models can be inserted by the aforementioned plug-

in mechanism. It is possible to use different models with different penalties in one segmentation. For instance, this may be used to get a lower number of parameters when a constant regression is nearly as good as a linear one. This might be useful for compression.

Both, hierarchic and dyadic segmentations, can be applied to images with arbitrary not necessarily dyadic dimensions. Images can be loaded and stored in a 16 Bit portable greymap format. Additionally, various formats can be imported. The segmentation algorithms use mainly floating point arithmetics with double precision (64 bit). This is also used for the cumulative sum matrices.

The software has an object oriented design, it is modular, portable and consists of readable and reusable code. There is a textual command based and a graphical user interface.

Besides the plots that are already provided by the software packages Voyager and ANTS$_{\text{InFields}}$, we have provided consistent interactive visualizations of the hierarchic and the dyadic partitions. This implies that the user gets immediate feedback for all computations. We believe in general that interactive visualization is one of the essential tools for the illustration of complex mathematical implications.

All objects implemented are fully persistent, which in the case of the dyadic segmentations has an interesting effect: When a quad tree plot is stored to disk, the relevant information contained in the complete tree is written. Opening the stored data from disk results in opening the tree plot, and all necessary information to reveal the image data for each parameter value γ is already contained. Large image data can be processed and regression results can be stored to disk in a first step. In a second step, the user may go on processing the minimization results for the whole stack of images provided by using different values of $\gamma \geq 0$. One application would be a presentation in a lecture: Demonstration panels, like the one displayed in Figure 6.4, can be prefabricated, opened and results are then displayed in realtime, compare Subsection 6.2.2.

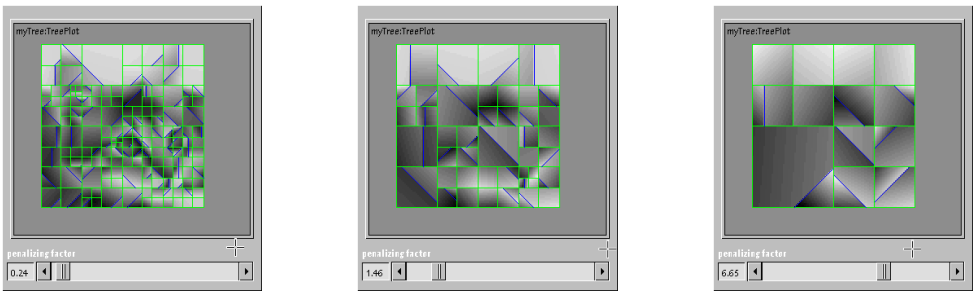


Fig. 6.4: Example of a persistent demonstration Panel. Minima are computed and visualized in realtime.

6.2 Runtime Analysis

Consider a set of angles Δ, data $z \in \mathbb{R}^S$ over a rectangle $S = \{1, \dots, N_1\} \times \{1, \dots, N_2\}$, a finite dimensional function space $\mathcal{F} \subset \mathbb{R}^S$ and a parameter $\gamma \geq 0$. According to Algorithm 4.2 the minimization of the Potts functional over wedge segmentations is performed in two steps:

(1) Allocation and computation of the cumulative sum matrices $I^{(\alpha)}(y)$ for all angles $\alpha \in \Delta$.

(2) Fast computation of a minimum $(\hat{\mathcal{P}}, \hat{f}_{\mathcal{P}})$ of $\|f_{\mathcal{P}} - z\|_2^2 + \gamma|\mathcal{P}|$ over wedge segmentations.

The following simulations have been carried out on a Windows variant of Oberon/Bluebottle (WinAos 1.04a). They were performed on a Pentium IV machine with 2.8 GHz and 1GB of memory. Test runtime measurements showed an increase in speed of the algorithms on the Linux variant of Oberon with a global speed-up factor of about 5/4. Native Oberon would be probably even a bit faster. The time resolution was about 15 ms.

6.2.1 Hierarchic Segmentation

For an analysis of the runtime, images with different sizes have been created, different models have been adopted, and the run times of steps (1)-(2) have been measured for each image and different models. Since the contents of the arrays do not influence the runtime, the images have been created with a uniform random grey value content. We have randomized the image sizes to prevent the runtime to be too dependent from a specific simulation scheme.

For example the garbage collection of a previous run could provide a systematic effect. Other side effects are the cache management depending on the underlying operating system, swap spaces in the memory etc. All simulations have been performed on a freshly started, clean environment.

Runtime versus Number of Pixels

For each model we have drawn 100 sample images with size $w \times w$, where w was drawn from a uniform distribution over $\{1, \ldots, 128\}$ for the rectangle models and $\{1, \ldots, 16\}$ for the wedge model. The following run times, corresponding to the minimization scheme in the previous paragraph, are displayed:
(1): Allocation of the cumulative sum matrices (blue circles)
(2): Minimization of the Potts functional (green crosses).

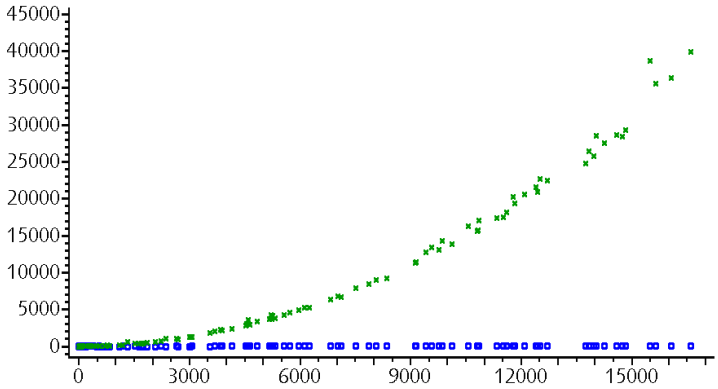


Fig. 6.5: Runtime (ms) versus number of pixels for a hierarchic rectangle segmentation with constant regression.

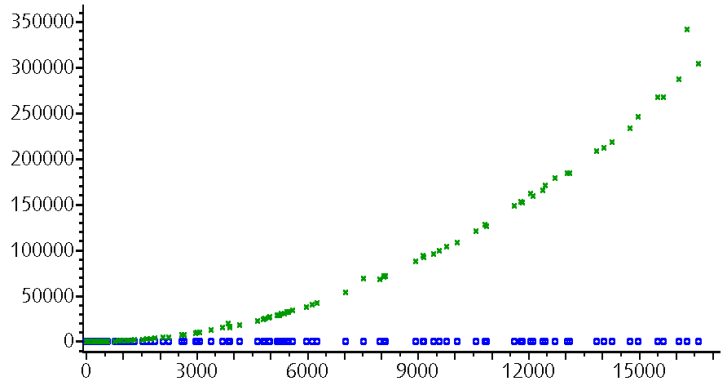


Fig. 6.6: Runtime (ms) versus number of pixels for a hierarchic rectangle segmentation with linear regression.

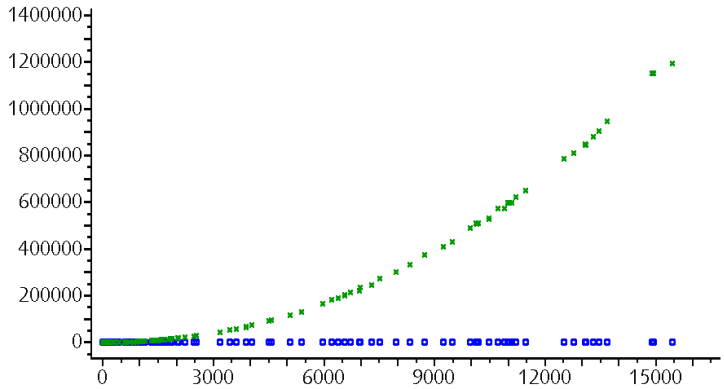


Fig. 6.7: Runtime (ms) versus number of pixels for a hierarchic rectangle segmentation with quadratic regression.

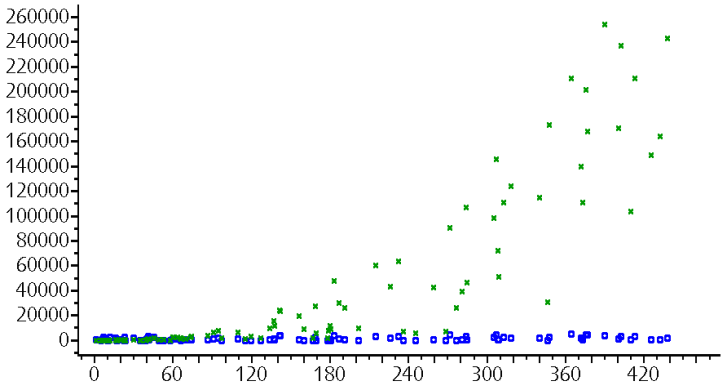


Fig. 6.8: Runtime (ms) versus number of pixels for a hierarchic wedge segmentation (4 angles) with constant regression.

Because the runtimes for the hierarchic model are very long for large images, we provided a mechanism to restrict the lengths of the segments in each dimension to a given set of lengths. For example, it is possible to perform the hierarchic wedge segmentation with dyadic lengths only. This results in reasonable runtimes for larger images. We have therefore performed runtime measurements with restriction to dyadic interval lengths as well. For each model, we have drawn 100 sample images with size $w \times w$ where w was drawn from a uniform distribution over $\{1, \ldots, 256\}$ and $\{1, \ldots, 128\}$.

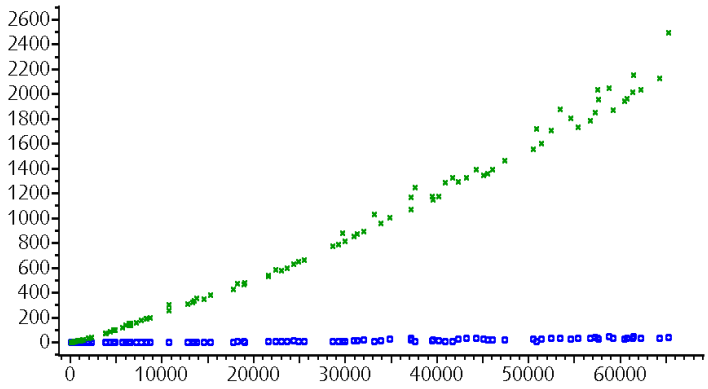


Fig. 6.9: Runtime (ms) versus number of pixels for a hierarchic rectangle segmentation with constant regression, restriction to dyadic interval lengths.

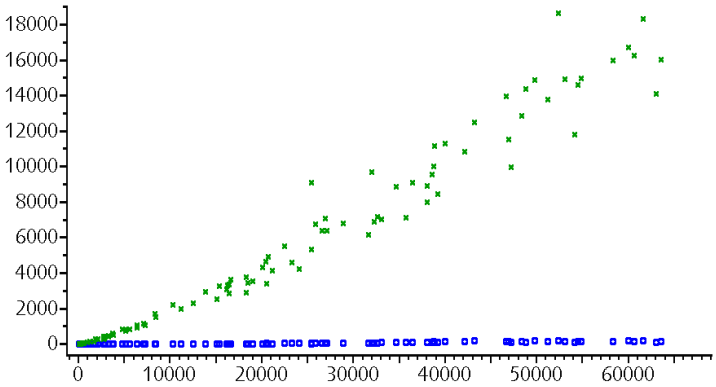


Fig. 6.10: Runtime (ms) versus number of pixels for a hierarchic rectangle segmentation with linear regression, restriction to dyadic interval lengths.

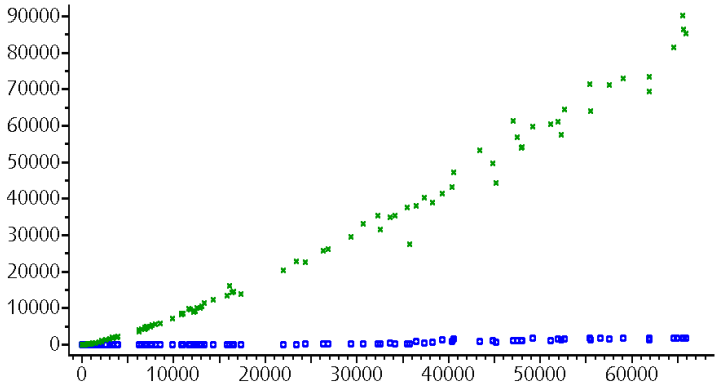


Fig. 6.11: Runtime (ms) versus number of pixels for a hierarchic rectangle segmentation with quadratic regression, restriction to dyadic interval lengths.

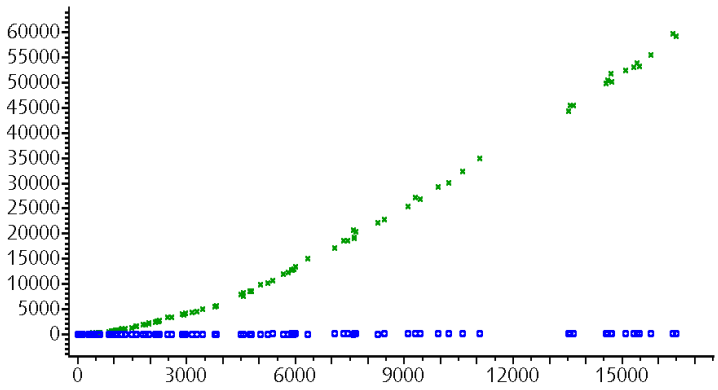


Fig. 6.12: Runtime (ms) versus number of pixels for a hierarchic wedge segmentation (4 angles) with constant regression, restriction to dyadic interval lengths.

Runtime versus Number of Angles

For a fixed image with dimensions 48×48 we have drawn 100 sample numbers n from a uniform distribution over $\{0, \ldots, 90\}$. For each number n a wedge segmentation with n angles has been computed. This runtime measurement has only been performed with restriction to dyadic interval lengths.

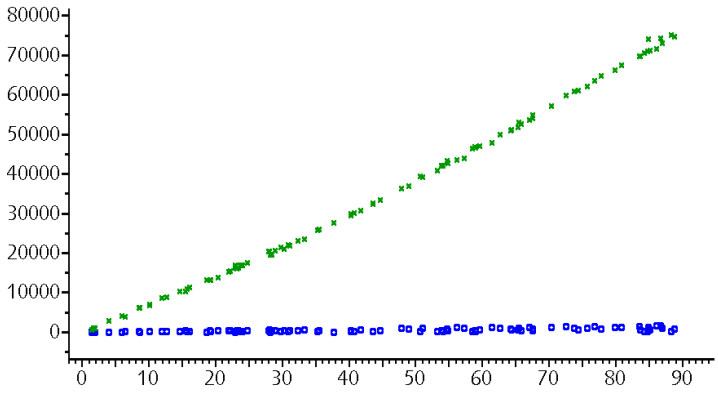


Fig. 6.13: Runtime (ms) versus number of angles for a hierarchic wedge segmentation with constant regression, restriction to dyadic interval lengths, image size 48×48.

6.2.2 Dyadic Segmentation

For dyadic wedge segmentations step (2) can be split into three steps since, by their tree structure, the local minimization results on each dyadic square can be stored in $O(|S|)$. Additionally the cumulative sum matrices have to be allocated for one angle only. This leads to the following minimization scheme.

(a) Allocate the cumulative sum matrices for one angle.

(b) Create a tree data structure according to Procedure CreateTree (p. 53).

(c) Repeat the following steps for all angles $\alpha \in \Delta$
- Compute the cumulative sum matrices for angle α.
- Perform the local minimization of $\|f_r - z\|_2^2$ in $f \in \mathcal{F}_r$ for each dyadic rectangle r according to Procedure LocalMin (p. 53). Keep the result in the tree if it is better than for the previous angles.

(d) Perform the minimization of the Potts functional with parameter γ by recursive minimization according to Procedure MinTree (p. 54).

The great advantages of this scheme are that firstly there is only a memory consumption of $O(|S| \cdot \dim \mathcal{F})$, which is crucial for large data, see next paragraph, and secondly after performing the preparation steps (1)-(3) **the minimization of the Potts functional can be performed in realtime for each** $\gamma \geq 0$. Once the preparation steps (a)-(c) have been performed

each result can be accessed more or less immediately. Therefore the result of the previous scheme can rather be seen as a **set of images** than one image for one parameter γ. We will make use of this result in the next section.

For an analysis of the runtime, images with different sizes have been created, different models have been adopted and the run times of steps (1)-(4) have been measured for each image and different models. All simulations have been performed on a freshly started, clean environment.

Runtime versus Number of Pixels

For each model we have drawn 1000 sample images with size $w \times w$ where w was drawn from a uniform distribution.

The following run times, corresponding to the minimization scheme in the previous paragraph, are displayed:
(a): Allocation of the cumulative sum matrices (blue framed rectangles)
(b): Allocation of the tree structure (black filled rectangles)
(c): Local minimization for all nodes in the tree (red crosses)
(d): Minimization of the Potts functional for $\gamma \geq 0$ (green filled rectangles).

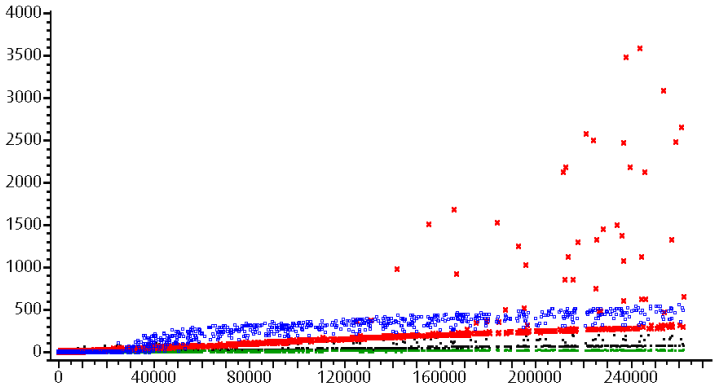


Fig. 6.14: Runtime (ms) versus number of pixels for a dyadic square segmentation with constant regression. 1000 sample images from $\{1 \times 1, \ldots, 512 \times 512\}$.

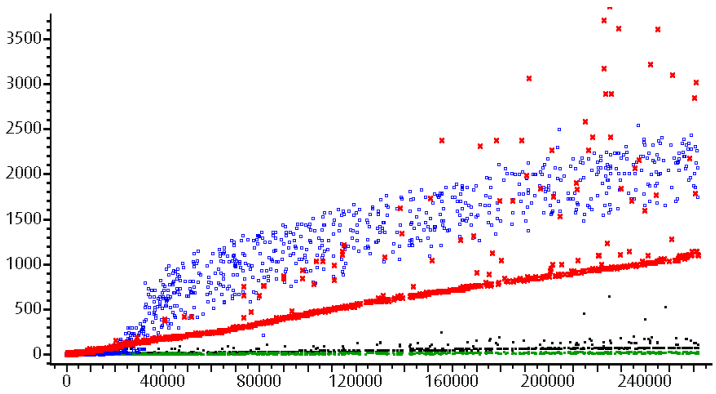


Fig. 6.15: Runtime (ms) versus number of pixels for a dyadic square segmentation with linear regression. 1000 sample images from $\{1 \times 1, \ldots, 512 \times 512\}$

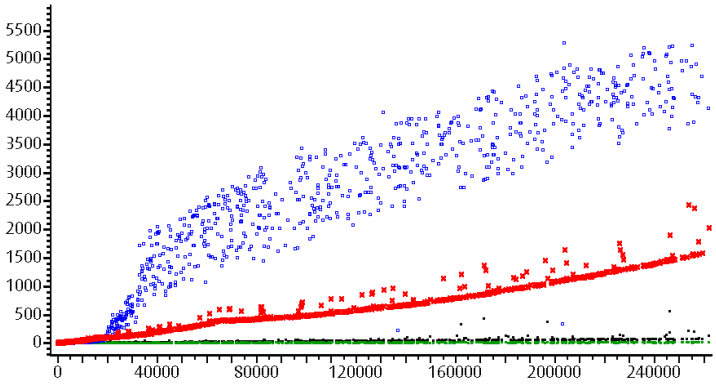


Fig. 6.16: Runtime (ms) versus number of pixels for a dyadic square segmentation with quadratic regression. 1000 sample images from $\{1 \times 1, \ldots, 512 \times 512\}$

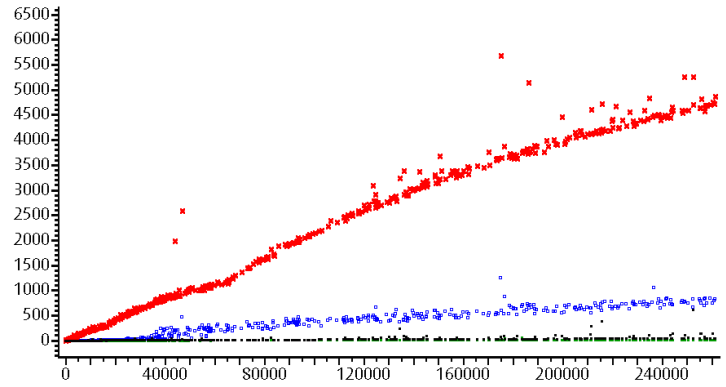


Fig. 6.17: Runtime (ms) versus number of pixels for a dyadic wedge segmentation (10 angles) with constant regression. 1000 sample images from $\{1 \times 1, \ldots, 512 \times 512\}$

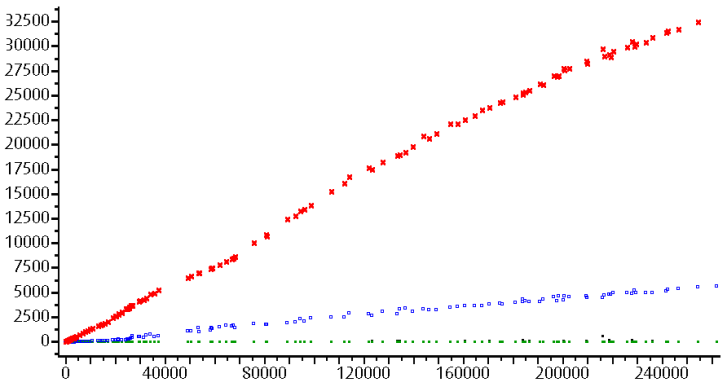


Fig. 6.18: Runtime (ms) versus number of pixels for a dyadic wedge segmentation (10 angles) with linear regression. 1000 sample images from $\{1 \times 1, \ldots, 512 \times 512\}$

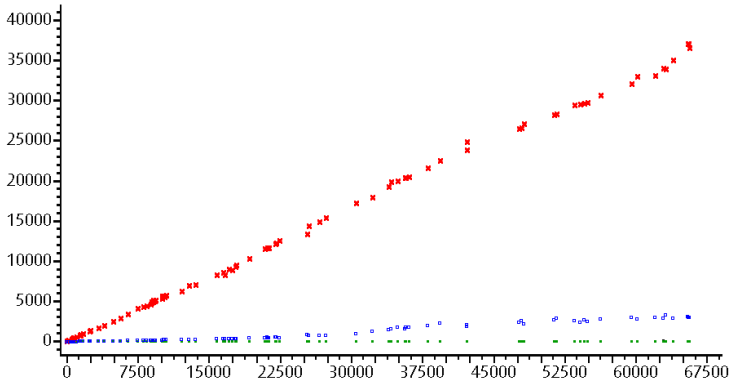


Fig. 6.19: Runtime (ms) versus number of pixels for a dyadic wedge segmentation (10 angles) with quadratic regression. 1000 sample images from $\{1 \times 1, \ldots, 256 \times 256\}$

Runtime versus Number of Angles

For images with fixed dimensions we have drawn sample numbers n from a uniform distribution over $\{0, \ldots, 359\}$. For each number n a wedge segmentation with n angles has been computed.

Displayed are the following run times corresponding to the minimization scheme:
(1): Allocation of the cumulative sum matrices (blue framed rectangles)
(2): Allocation of the tree structure (black)
(3): Local minimization for all nodes in the tree (red crosses)
(4): Minimization of the Potts functional for $\gamma \geq 0$ (green).

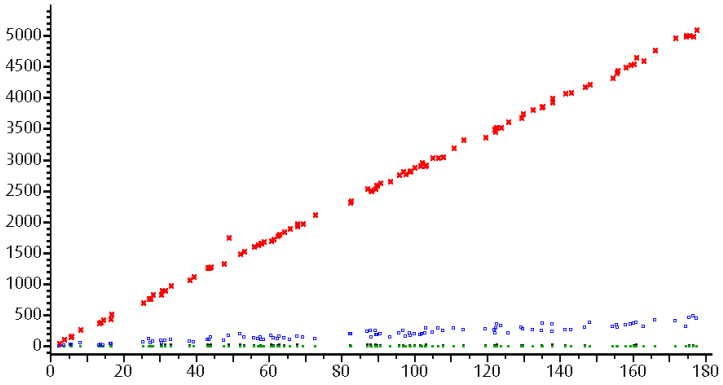


Fig. 6.20: Runtime (ms) versus number of angles for a piecewise constant dyadic wedge segmentation. 100 samples, image size 128×128.

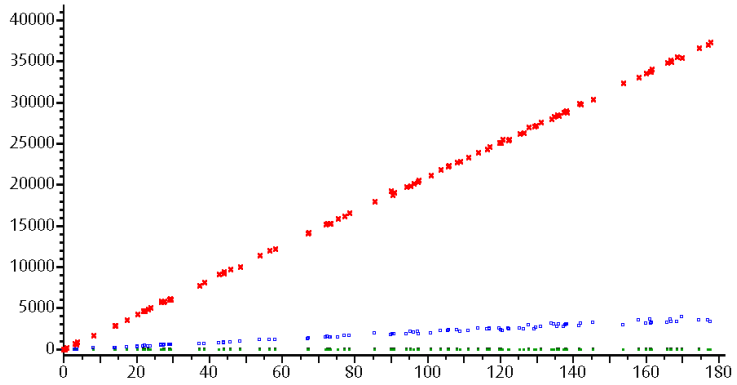


Fig. 6.21: Runtime (ms) versus number of angles for a piecewise linear dyadic wedge segmentation. 100 samples, image size 128×128.

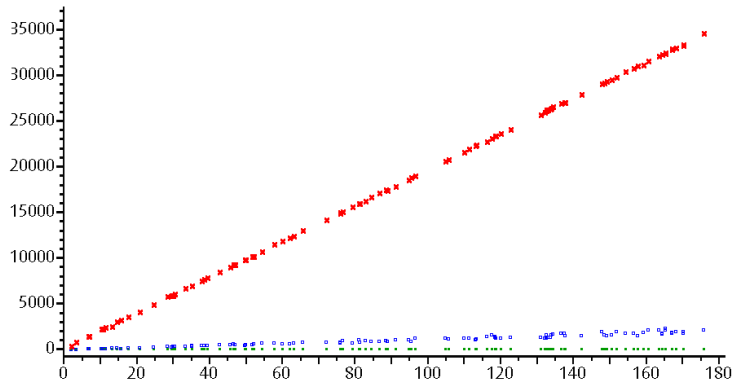


Fig. 6.22: Runtime (ms) versus number of angles for a piecewise quadratic dyadic wedge segmentation. 100 samples, image size 128×128.

The set of angles can be adaptively used: Half of the set of angles is used for each increasing depth of the corresponding quad tree. A motivation for this procedure is presented in Paragraph 6.3.3 below. Using this adaptive scheme provides the following runtime measurements.

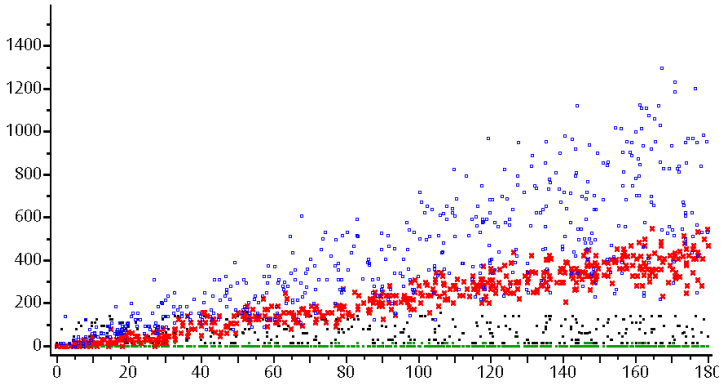


Fig. 6.23: Runtime (ms) versus number of angles for a piecewise constant dyadic wedge segmentation. 1000 samples, image size 128×128, adaptive angles.

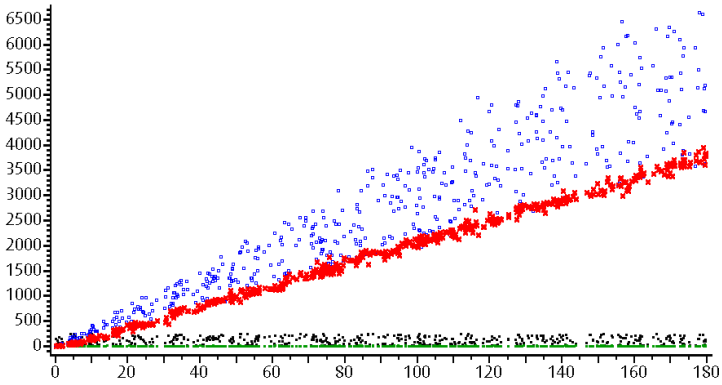


Fig. 6.24: Runtime (ms) versus number of angles for a piecewise linear dyadic wedge segmentation. 1000 samples, image size 128×128, adaptive angles.

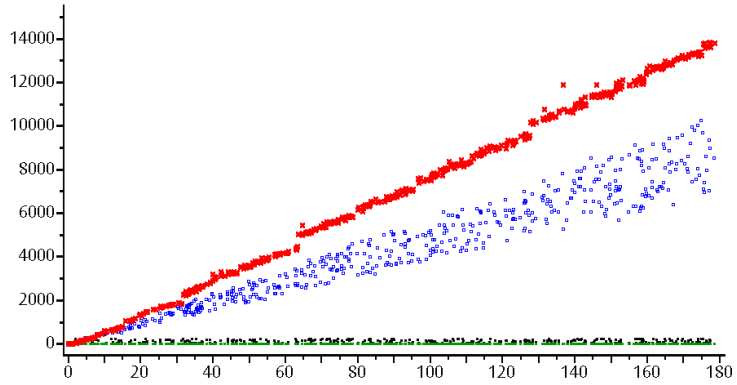


Fig. 6.25: Runtime (ms) versus number of angles for a piecewise quadratic dyadic wedge segmentation. 1000 samples, image size 128×128, adaptive angles.

6.2.3 Comparison with 'BeamLab'

The extraordinary fast runtimes of our software can be shown by comparison with another library for computation of wedge segmentations, 'BeamLab'. BeamLab is a package running on MatLab. It contains routines for constant wedgelet segmentations of images with exclusively dyadic dimensions. We used MatLab version 6 and BeamLab as loaded from the webpage on the 8.6.2004, see Donoho et al. (2004). For a wedgelet segmentation BeamLab needed on a 2.8 GHz Pentium IV machine the following runtimes dependent on the image sizes. For a comparison we append approximate runtimes for a wedgelet model in our approach with different number of angles. An adaptive model removes half of the angles when entering the next depth in the quad tree. The 'equivalent' model is an adaptive model with a number of 1024, 512, ..., 32 angles corresponding to image sizes 512×512, 256×256, ..., 16. This should be roughly equivalent with considering every wedgelet in the image.

Image size	16×16	32×32	64×64	128×128	256×256	512×512
BeamLab	$6.72s$	$45.34s$	$330.41s$ $\approx 5.5min$	$2676.3s$ $\approx 44.6min$	$27918s$ $\approx 7h45min$	$> 12h$
dyadic wedge equivalent	$< 0.02s$	$0.05s$	$0.2s$	$1.2s$	$8s$	$59s$
dyadic wedge 1024 angles	$1s$	$2s$	$8s$	$37s$	$140s$	$722s$
dyadic wedge 180 angles	$0.2s$	$0.3s$	$1.6s$	$6s$	$24s$	$120s$
dyadic wedge 4 angles	$< 0.01s$	$< 0.01s$	$0.04s$	$0.2s$	$1s$	$10s$
dyadic square	$< 0.01s$	$< 0.01s$	$< 0.01s$	$0.04s$	$0.5s$	$6s$

6.2.4 Synopsis

The computation of hierarchic segmentations is in general much slower than the computation of dyadic partitions.

The simulations have experimentally validated Theorem 4.3.1. The runtime is linear in the number of angles. It is quadratic in the number of pixels in the case of the hierarchic model, and it is linear in the number of pixels for the dyadic segmentations. If only dyadic interval lengths are permitted in the hierarchic model, it shows a runtime behavior that is close to linear. Concerning the number n of functions in the constant, linear and quadratic model, $n = 1$, $n = 3$ and $n = 6$, the runtime simulation showed a much less than n^3 behavior. This was expected since the computational overhead for the tree traversal and the computation of moments with the cumulative sum matrices, which is $O(n^2)$, dominates the procedure for relatively small dimensions of the local regression model.

The hierarchical model has a good theoretical performance in special cases such as the approximation of horizon functions, compare Paragraph 5.3.4. However, its large runtimes for reasonable image data makes it unsuitable in practical applications. A restriction of the considered stripe widths and heights to, for instance, dyadic values makes its performance more tolerable. However, since the method seems in most cases to be inferior to the dyadic approach, we content ourselves to show one or two results in the next paragraph and will then stop to address the hierarchic segmentations.

For dyadic square segmentations the main part of the runtime was needed for the allocation and computation of the cumulative sum matrices. For dyadic wedge segmentations the runtime consisted mainly of the computation time of the local regression part. The time for the allocation of the tree and the

global minimization can be neglected in comparison with the other parts. Once the preparation steps (1)-(3) have been performed, the minimization of the Potts functional can be done more or less immediately, the plots show that this time is about the time resolution (0-30 ms). Interestingly the dyadic wedge segmentations became faster the longer the simulation ran (not visible in the plots). We suppose that this is due to a reorganization of the memory structure resulting in a faster access to local variables in the recursive procedures.

Compared to an old version of our software (as cited in Wicker (2004)) we could increase the speed by a factor of about 3. A comparison to the package BeamLab yields a speed-up factor of about 3500.

6.3 First Experiments: Phenomenological Description

In this section we will comment on the properties of wedge segmentations when applied to natural images or phantoms. We start with an analysis of comparability when sizes or grey-value ranges of images are scaled. Then we present a collection of concrete examples with different types of image data. We close this section with some experimental results concerning the angular resolution of wedge segmentations and a short experiment for horizon functions.

6.3.1 Scaling of γ

In order to compare segmentations of images with different size or grey-value range the segmentations should be 'nearly invariant' under the respective scaling operations. We make this more precise in this paragraph. Recall that we consider the minimization of the functional

$$H(z, (\mathcal{P}, f_{\mathcal{P}})) = \gamma \cdot |\mathcal{P}| + \|f_{\mathcal{P}} - z\|_2^2$$

for segmentations $(\mathcal{P}, f_{\mathcal{P}})$ from some segmentation class $\mathfrak{S}$ over an image domain S.

For notational simplicity we consider squares $S_n = \{1, \ldots, n\}^2$, $n \in \mathbb{N}$. Recall the scaling operators $\boldsymbol{\iota}^{(n)}$, $n \in \mathbb{N}$, defined in Chapter 5 on page 125 that embed segmentations over S_n into the continuous square $[0, 1)^2$.

Lemma 6.3.1: *Consider for each $n \in \mathbb{N}$ a class of segmentations $\mathfrak{S}_n$ over S_n. Let for each $n \in \mathbb{N}$*

$$\mathfrak{F}_n = \{\boldsymbol{\iota}^{(n)}(\mathcal{P}, f_\mathcal{P}) : (\mathcal{P}, f_\mathcal{P}) \in \mathfrak{S}_n\}.$$

Let $n_1, n_2 \in \mathbb{N}$, $z_1 \in \mathbb{R}^{S_{n_1}}, z_2 \in \mathbb{R}^{S_{n_2}}$ and assume that $\boldsymbol{\iota}^{(n_1)}(z_1) = \boldsymbol{\iota}^{(n_2)}(z_2)$ and that there is a segmentation class $\mathfrak{S}$ over $[0,1)^2$ such that $\mathfrak{S} = \mathfrak{F}_{n_1}$ and simultaneously $\mathfrak{S} = \mathfrak{F}_{n_2}$. Let $\gamma > 0$. Then, with $\gamma_1 = |S_1| \cdot \gamma$ and $\gamma_2 = |S_2| \cdot \gamma$, the following holds:

$$\boldsymbol{\iota}^{(n_1)}\left(\operatorname*{argmin}_{(\mathcal{P}, f_\mathcal{P}) \in \mathfrak{S}_1} H_{\gamma_1}(z_1, (\mathcal{P}, f_\mathcal{P}))\right) = \boldsymbol{\iota}^{(n_2)}\left(\operatorname*{argmin}_{(\mathcal{Q}, f_\mathcal{Q}) \in \mathfrak{S}_2} H_{\gamma_2}(z_2, (\mathcal{Q}, f_\mathcal{Q}))\right).$$

Proof. Since by definition of $\boldsymbol{\iota}^{(n)}$ and $\boldsymbol{\delta}^{(n)}$ it holds that $\boldsymbol{\delta}^{(n)}(\boldsymbol{\iota}^n(z))$ for all $z \in \mathbb{R}^{S_n}$, we can apply Lemma 5.1.7 and get

$$(\mathcal{P}, f_\mathcal{P}) \in \operatorname*{argmin}_{\mathfrak{S}_1} H_\gamma^{(n_1)}(z_1, \cdot)$$

if and only if

$$\boldsymbol{\iota}^{n_1}(\mathcal{P}, f_\mathcal{P}) \in \operatorname*{argmin}_{\mathfrak{S}} \tilde{H}_\gamma^{(n_1)}(\boldsymbol{\iota}^{(n_1)}(z_1), \cdot)$$

and the respective result for $\mathfrak{S}_2$. Since the minimizers of $H_\gamma^n(z, \cdot)$ are the minimizers of $H_{|S_n| \cdot \gamma}(z, \cdot)$ the proof is complete. □

Lemma 6.3.1 implies that differently sized versions of the same data generate the same segmentations up to size, if γ is scaled by the size, provided that the segmentation classes are the same up to a scaling. In practice this assumption about the segmentation classes is hardly ever fulfilled but it is a good approximation and experiments showed that the segmentations look very similar. See Wicker (2004), page 94. It is easy to see that the statement of the previous lemma is also true for arbitrary rectangular image domains.

Lemma 6.3.2: *Consider an image domain S, data $z \in \mathbb{R}^S$ and a real value $\gamma > 0$. Let a class of segmentations $\mathfrak{S}$ over S be given such that $(\mathcal{P}, f_\mathcal{P}) \in \mathfrak{S}$ implies $(\mathcal{P}, \mu \cdot f_\mathcal{P}) \in \mathfrak{S}$ for all $\mu > 0$. Let $\gamma > 0$. Then*

$$(\hat{\mathcal{P}}, \hat{f}_\mathcal{P}) \in \operatorname*{argmin}_{\mathfrak{S}} H_{\lambda^2 \gamma}(z, \cdot)$$

implies for all $\lambda > 0$ that

$$(\hat{\mathcal{P}}, \hat{f}_\mathcal{P}/\lambda) \in \operatorname*{argmin}_{\mathfrak{S}} H_\gamma(z/\lambda, \cdot).$$

Proof. It holds that $\|z - f_{\mathcal{P}}\|_2^2/\lambda^2 = \|z/\lambda - f_{\mathcal{P}}/\lambda\|_2^2$. Therefore a location of a minimum $(\hat{\mathcal{P}}, \hat{f}_{\mathcal{P}})$ of $(\mathcal{P}, f_{\mathcal{P}}) \mapsto \gamma \cdot \lambda^2|\mathcal{P}| + \|z - f_{\mathcal{P}}\|_2^2$ is a minimizer of $(\mathcal{P}, f_{\mathcal{P}}) \mapsto \gamma|\mathcal{P}| + \|z/\lambda - f_{\mathcal{P}}/\lambda\|$ and therefore $(\hat{\mathcal{P}}, \hat{f}_{\mathcal{P}}/\lambda)$ minimizes

$$(\mathcal{P}, f_{\mathcal{P}}) \mapsto \gamma|\mathcal{P}| + \|z/\lambda - f_{\mathcal{P}}\|.$$

□

The previous lemma implies that different (grey-)value-scaled versions of one image produce the same segmentation up to a scaling of the function values if γ is scaled with the squared scaling factor. In practice this scaling parameter λ is the number of grey-values.

We conclude that for a good comparison of outcomes of the minimization for data z with different geometries and (grey-)value range, the parameter γ has to be scaled with the size of image data and with the square of numbers of greyvalues λ. All images and experimental data in this thesis have been created with the scaled parameter

$$\gamma_{scaled} := \gamma \cdot |S| \cdot \lambda^2/(256)^2.$$

6.3.2 Visual Inspection

In this section we present a collection of concrete examples produced by the methods developed in this thesis. We selected the examples in order to illustrate what kind of result the algorithms produce with different types of image data. The data are representative for some important image features, like texture structure, boundaries, contrast, edges etc. There is no rigorous treatment but only a phenomenological analysis. We evaluate the estimates by visual inspection. Nevertheless this should give us valuable hints for the performance of the algorithms.

Example 6.3.3 ('new york'): *In Figure 6.26 various outcomes of the hierarchic model are displayed. Since the hierarchic segmentation is much slower than the dyadic segmentation, for complex models the outcome was restricted to dyadic lengths only.*

Example 6.3.4 ('patterns'): *In Figure 6.27 a collection of Brodatz textures is displayed. The figure shows that a wedge segmentation cannot resolve*

patterns. The relatively good reconstruction of parts of the image comes from the quadratic regression model.

Example 6.3.5 ('bird'): *Figure 6.28 demonstrates the adaptivity of the quad tree model. The relevant large scale structures in the image are well detected.*

Example 6.3.6 ('barbara'): *Figure 6.29 demonstrates that the dyadic wedge model with linear regression is well suited to detect fine structure with high contrast. Observe how the black and white stripes in the image are well approximated by the model.*

Example 6.3.7 ('contrast'): *Figure 6.30 shows that the L^2-distance of the Potts functional leads to different treatment of same geometric structures with different contrast to the background in the image.*

Example 6.3.8 ('chess'): *In Figure 6.31 a 5×5- chess board is segmented with a dyadic square partition. Since the board is not dyadic, the dyadic partition shows artefacts while the hierarchic partition does a perfect reconstruction with far less segments. The artefacts arise since the quad tree structure is not shift invariant. The double lines on the hierarchic partition are caused by grey-values in the original image.*

Example 6.3.9 ('K'): *In Figure 6.32 a typical artefact of the linear model is displayed. It appears already in usual one-dimensional linear regression. Strong contrasts in image data lead to big gradients in the rectangle. Because of linearity, they are balanced by an overdrawing at the smooth edge of a wedge and new discontinuities are produced.*

Example 6.3.10 ('rotation'): *Figure 6.33 displays a typical case where a wedge model is superior to the dyadic model. Additionally it shows that the wedge model provides quad tree artefacts where an angle of some edge in the image does not exactly fit to the model.*

Example 6.3.11 ('ibb'): *In Figure 6.34 it can be observed that the linear wedge model is well suited for smooth color transitions. The corresponding edge plot shows that the linear model needs less squares to represent more or less the same result.*

Example 6.3.12 ('slope'): *Figure 6.35 displays the difference between a segmentation with constant and one with linear regression. Smooth color transitions are well recognized by the linear model. Additionally one can see that the linear model has a direction and therefore performs well with less wedge intersection than in the constant case. Again quad tree artefacts and different contrast levels lead to irregular approximation.*

Example 6.3.13 ('splash'): *Figure 6.36 shows that dependent on the parameter γ structures with very little contrast such as noise and the mirroring of the splash in the water are removed. This indicates that the segmentation might be useful for denoising of images.*

Example 6.3.14 ('zebra'): *In Figure 6.37 the dyadic wedge model is compared with a sophisticated edge detection algorithm from stochastic image analysis. It has been performed using* ANTS$_{InFields}$ *and is explained in Winkler (2002), p. 223. Interestingly the relatively simple dyadic wedge segmentation extracts similar features as the stochastic algorithm.*

Example 6.3.15 ('goldhill' and 'boat'): *In Figures 6.38 and 6.39 the parameter dependence of the outcome of the algorithm is displayed. A low value of γ obviously corresponds to a lot of structure and many segments while larger values result in smoother images with less information needed for a specification.*

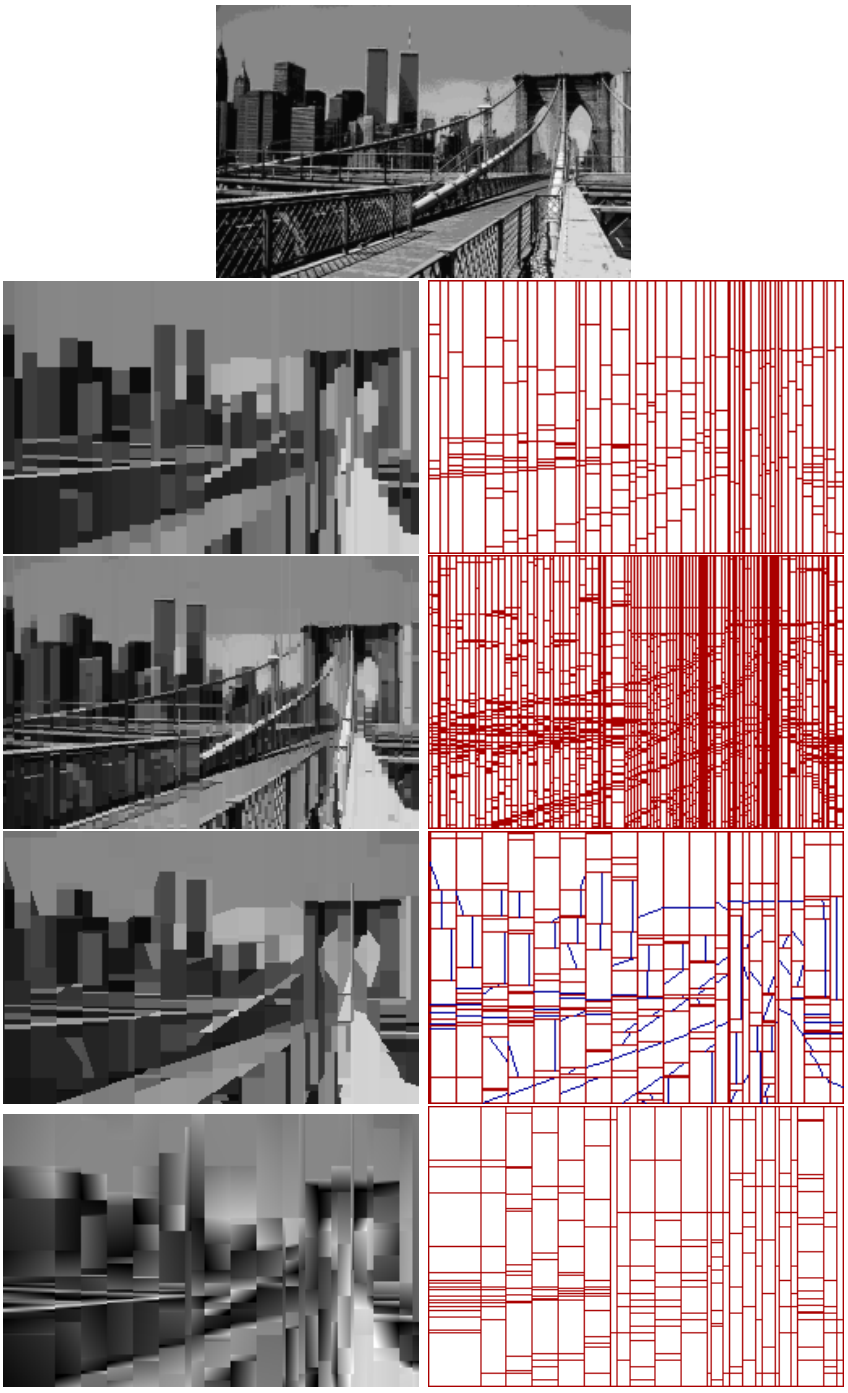


Fig. 6.26: Image 'nyc', 256 × 168 (top), segmentations with a hierarchic model. Constant regression with no limitations, $\gamma = 1$; constant regression with dyadic lengths, $\gamma = 0.1$; wedge model with constant regression, dyadic lengths, 10 angles, $\gamma = 1$; linear regression, dyadic lengths (from top to bottom).

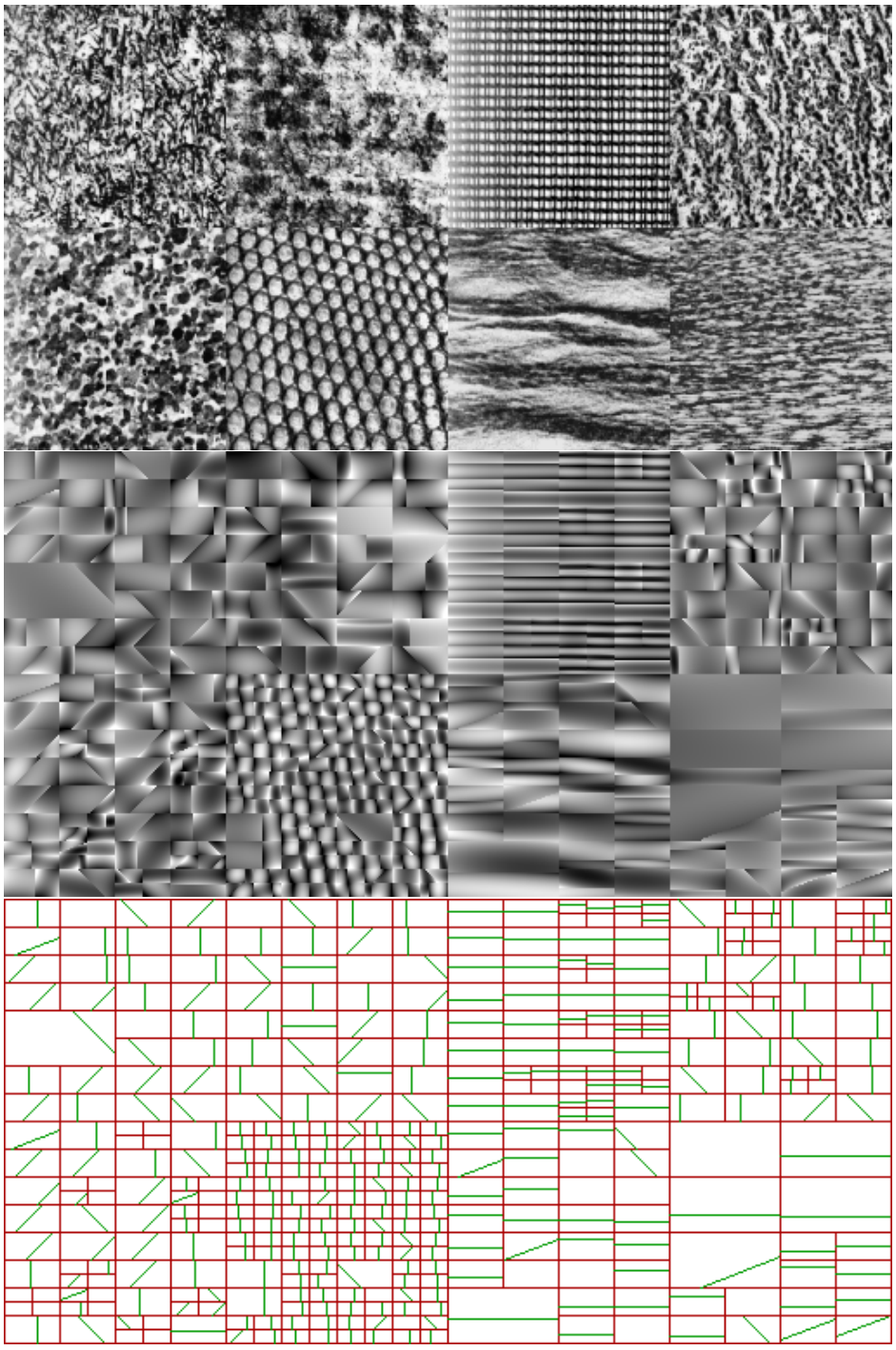


Fig. 6.27: Image 'patterns', 512×256 (top), segmentation with a dyadic square and a dyadic wedge model (5 angles) with quadratic regression. $\gamma = 0.2$ (middle and bottom).

Fig. 6.28: Image 'bird', 256×256, segmentation with a dyadic square model with constant regression. $\gamma = 0.01$

Fig. 6.29: Image 'barbara', 256×256, segmentation with a dyadic square model and a wedge model (180 angles) with linear regression. $\gamma = 0.002$

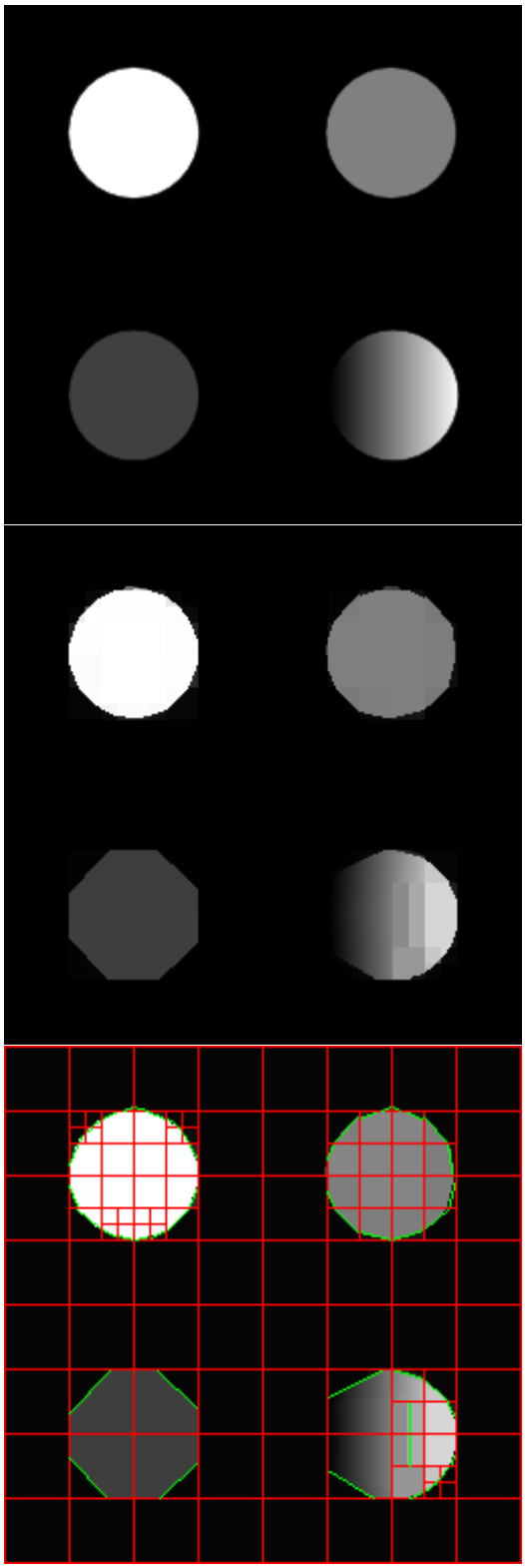


Fig. 6.30: Image 'contrast', 256×256, segmentation with a dyadic square model and dyadic wedge model (180 angles, adaptive with offset 2) with constant regression and linear regression. $\gamma = 0.34$

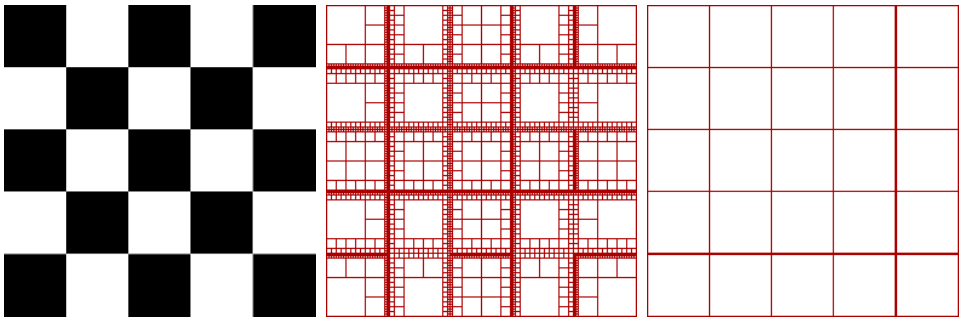


Fig. 6.31: Image 'chess', 256 × 256, segmentation with a dyadic square model with constant regression and with a hierarchic model with constant regression. $\gamma = 0.01$

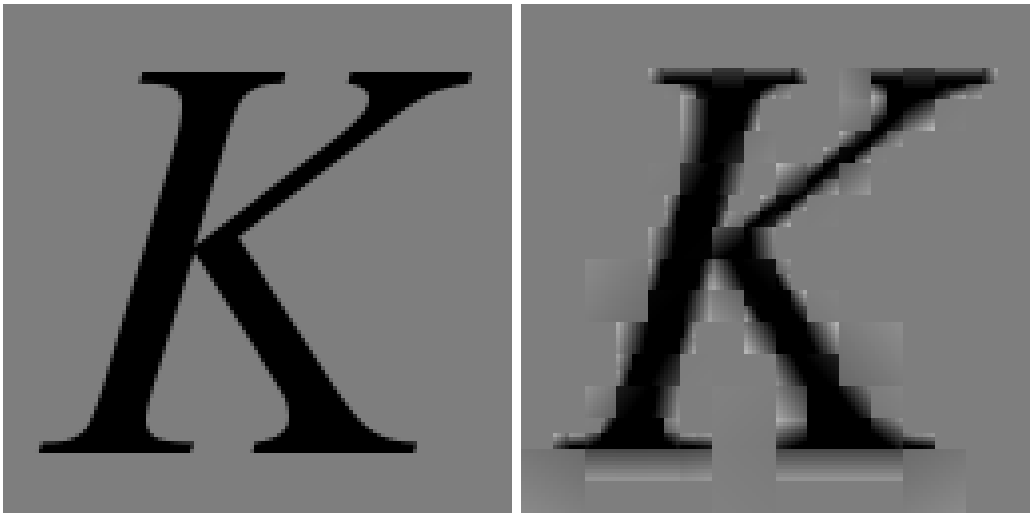


Fig. 6.32: Image 'K', 128 × 128, segmentation with a dyadic square model with linear regression. $\gamma = 0.27$

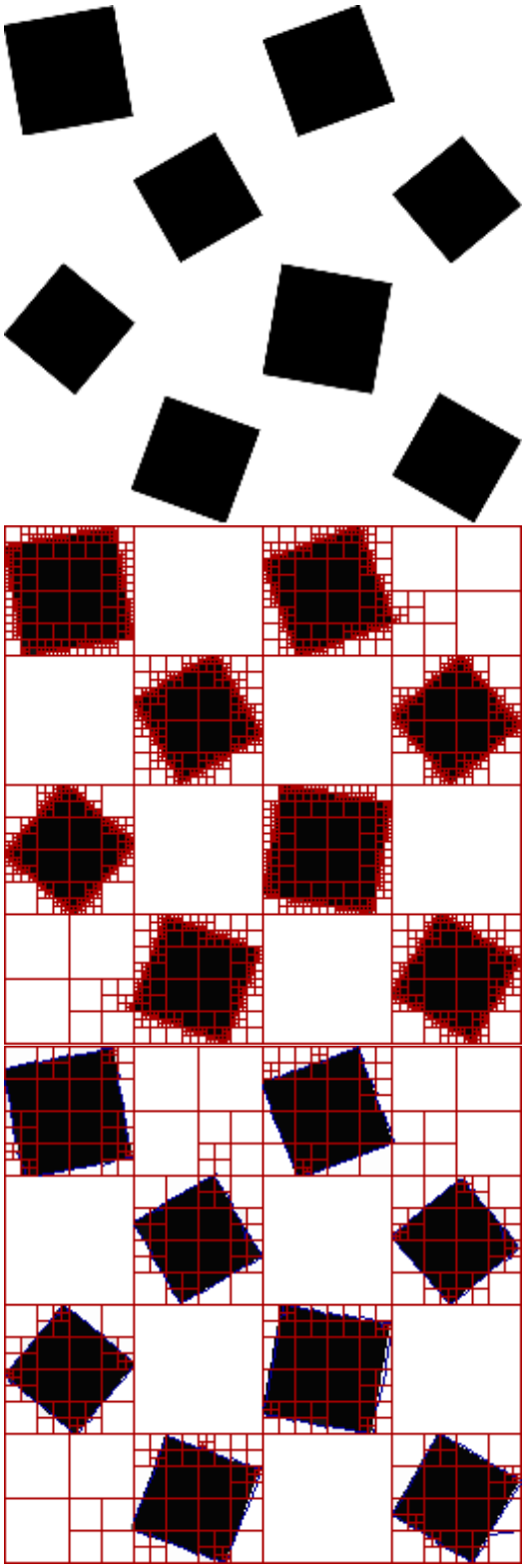


Fig. 6.33: Image 'rotation', 256×256, segmentation with a dyadic square model and dyadic wedge model (180 angles) with constant regression. $\gamma = 0.1$

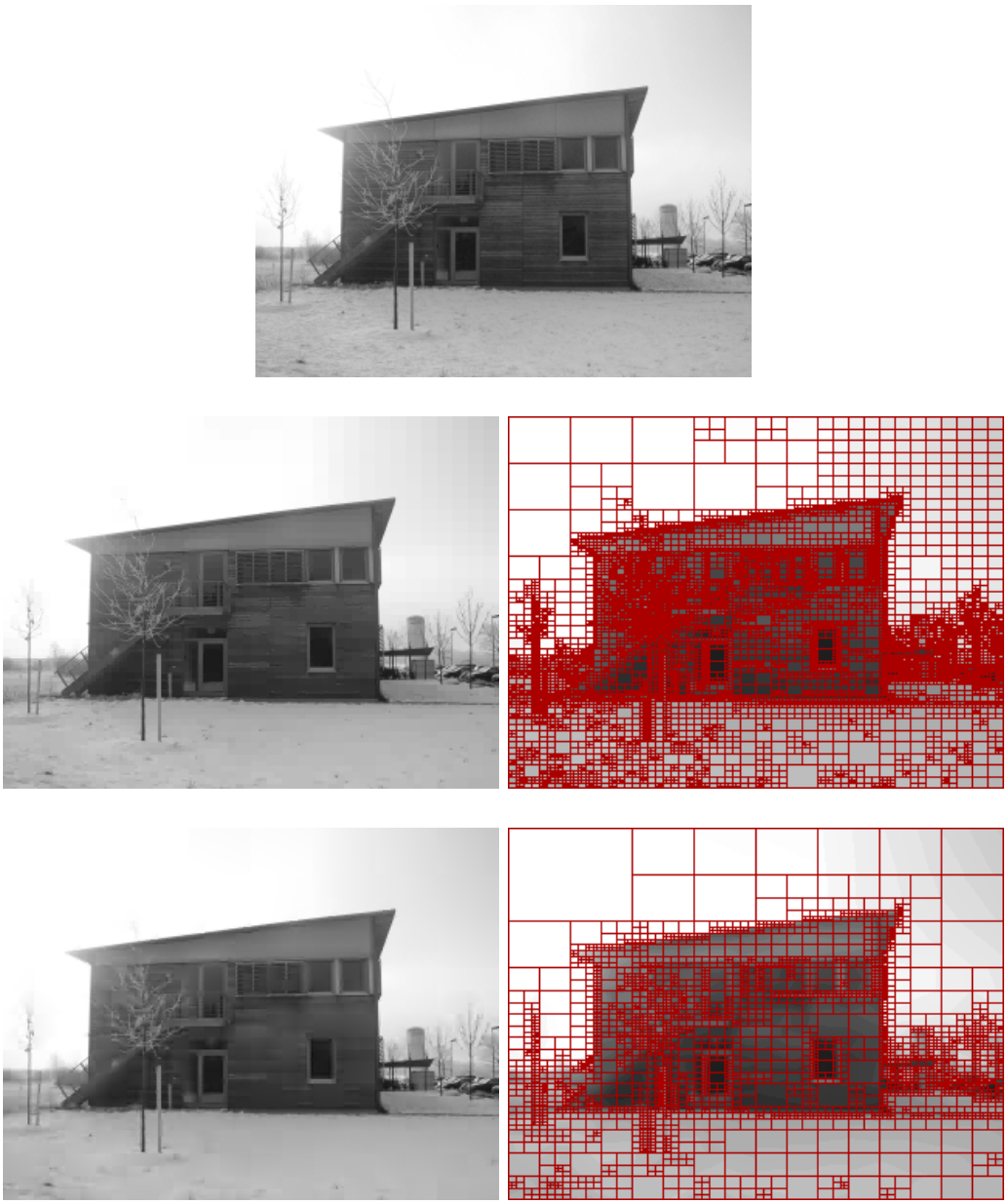


Fig. 6.34: Image 'ibb north', $324x240$, segmentation with a dyadic square model with constant regression (middle) and linear regression (bottom). $\gamma = 0.01$

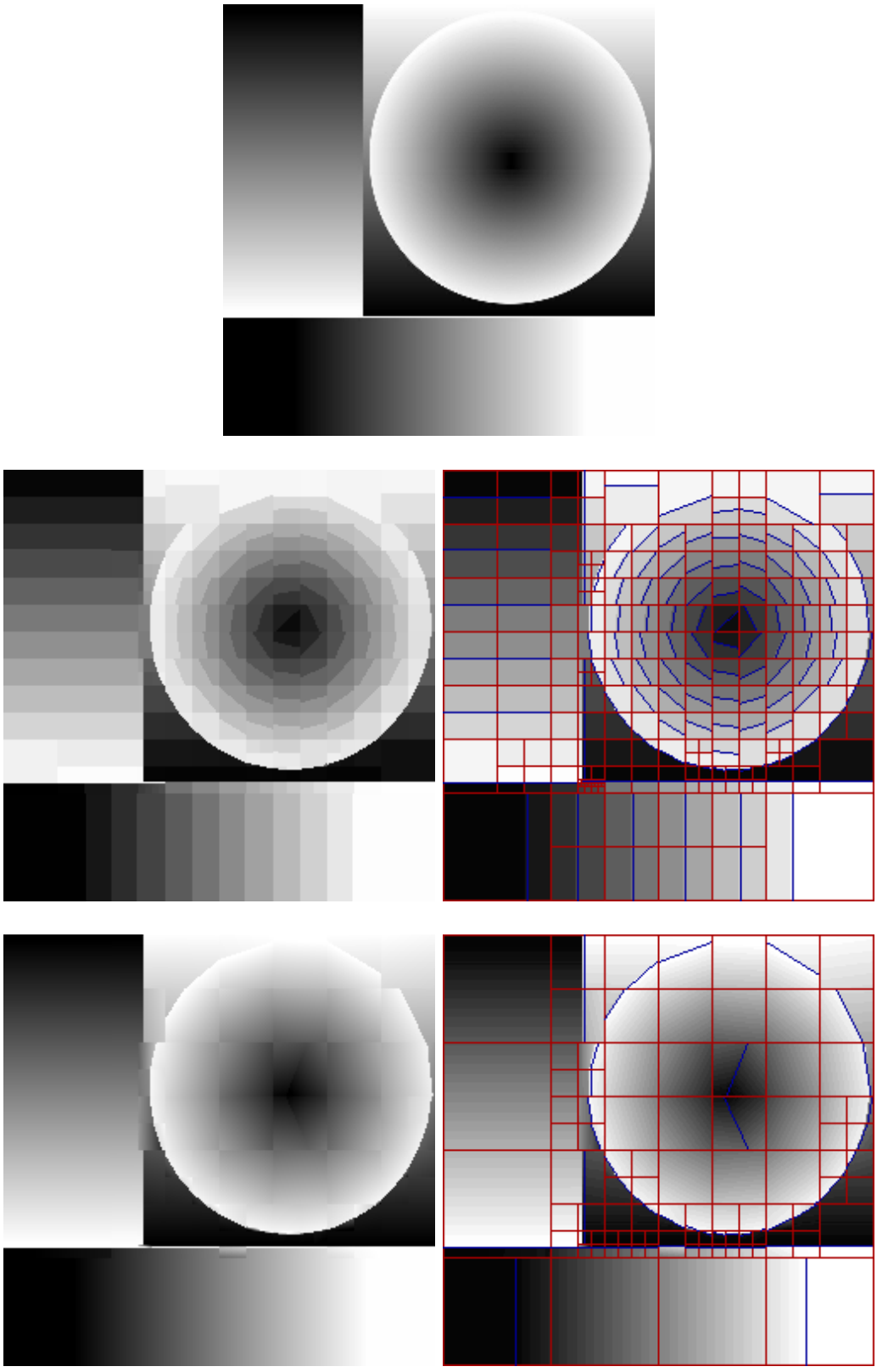


Fig. 6.35: Image 'slope', 256×256, segmentation with a dyadic wedge model (180 angles) with constant regression, $\gamma = 0.25$ (middle) and linear regression (bottom), $\gamma = 0.17$.

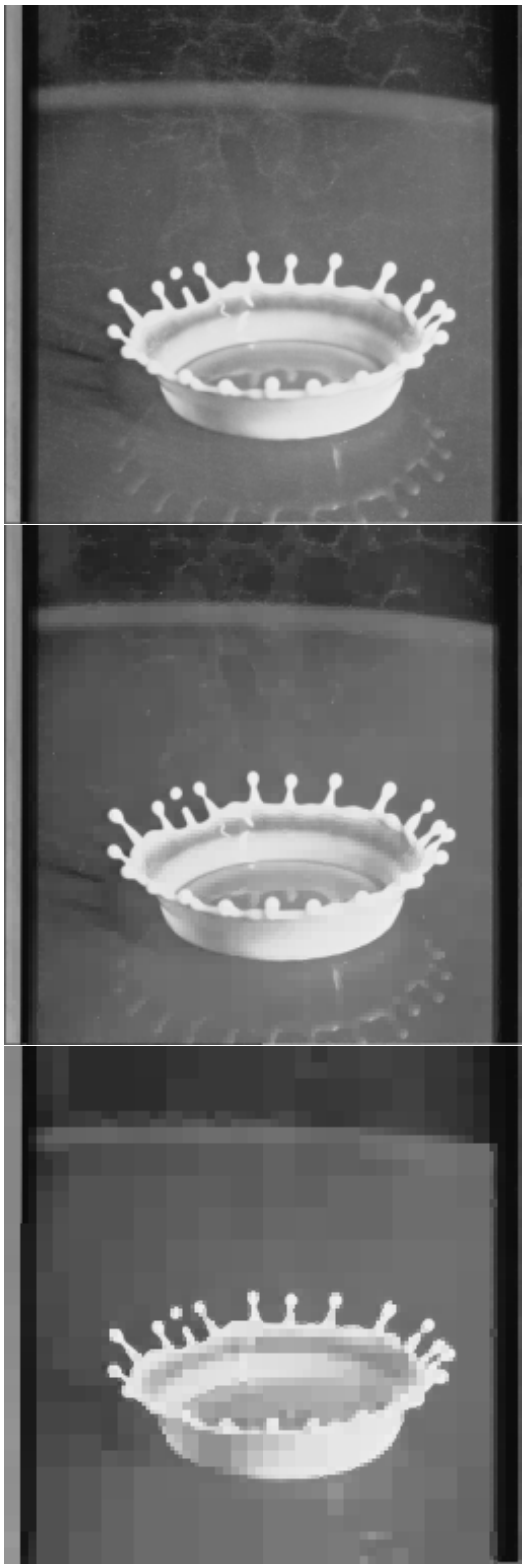


Fig. 6.36: Image 'splash', 256×256, segmentation with a dyadic square model with constant regression, $\gamma = 0.001$ (middle) and $\gamma = 0.03$ (bottom).

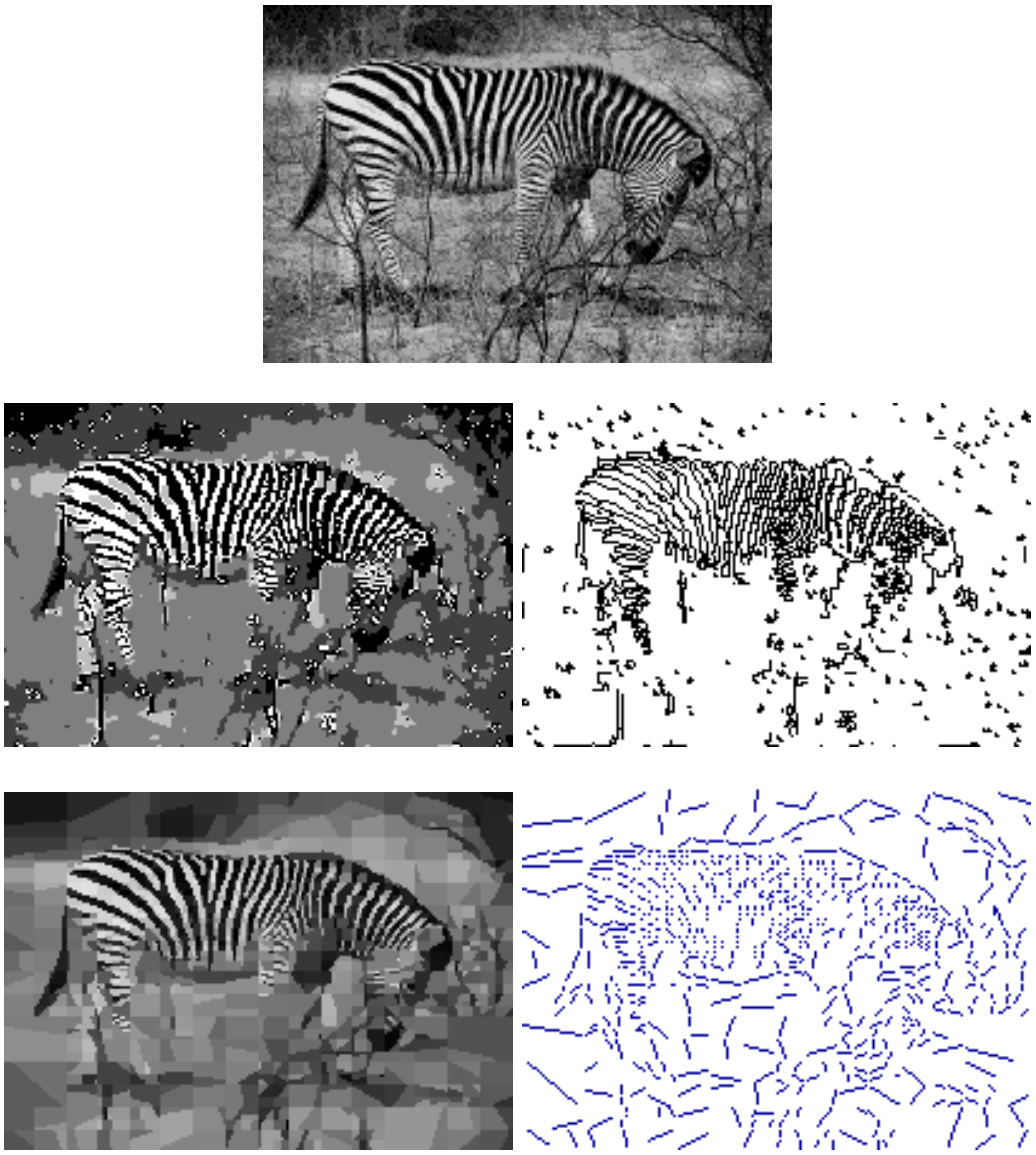


Fig. 6.37: Image 'zebra', 210×142, pixel edge process sampled with a Gibbs sampler (middle), segmentation with a dyadic square model and wedge Model (180 angles) with constant regression, $\gamma = 0.26$ (bottom). For a better comparability only wedges are displayed in the partition plot.

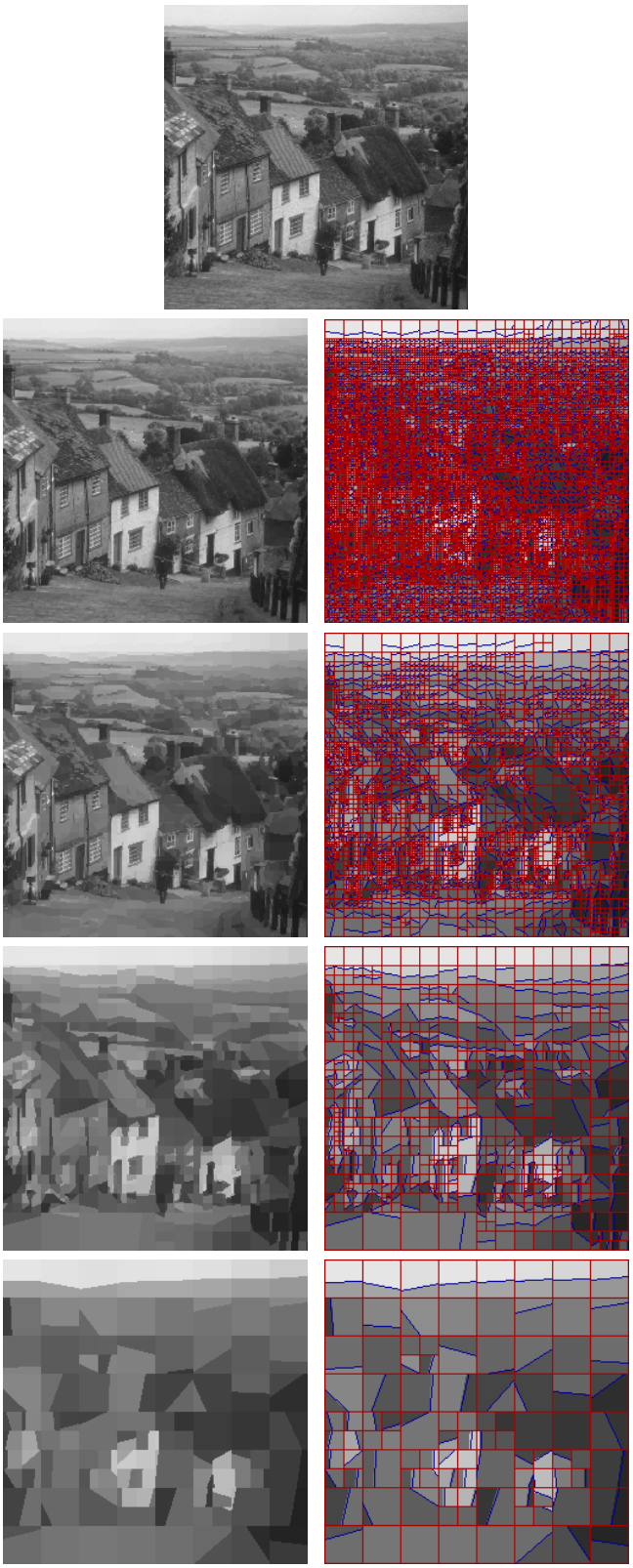


Fig. 6.38: Image 'goldhill', 256×256, segmentations with a dyadic square model and wedge Model (180 angles) with constant regression, $\gamma = 0.001$, $\gamma = 0.01$, $\gamma = 0.1$ and $\gamma = 1$ (from top to bottom).

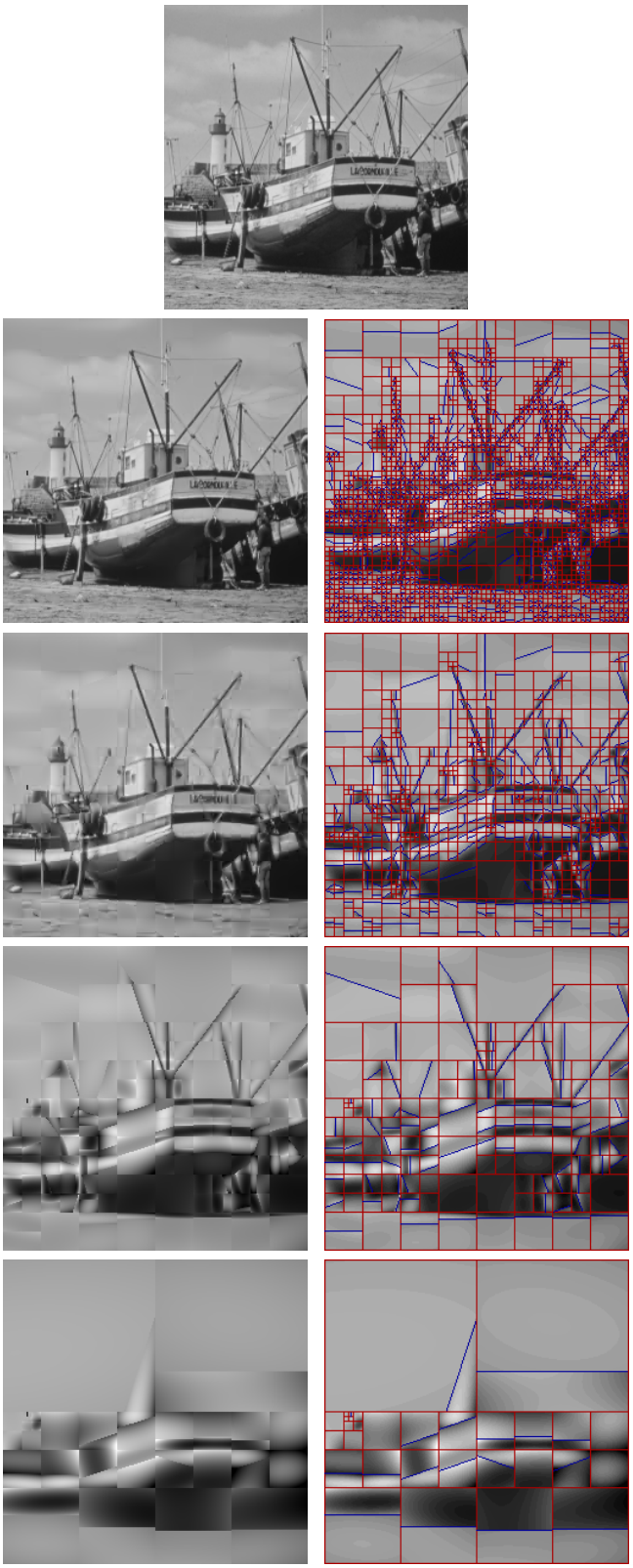


Fig. 6.39: Image 'boat', 256 × 256, segmentations with a dyadic square model and wedge Model (180 angles) with quadratic regression, $\gamma = 0.001$, $\gamma = 0.01$, $\gamma = 0.1$ and $\gamma = 1$ (from top to bottom).

6.3.3 Angular Resolution

A well known measure to rate the quality of approximations of images is the Peak to Signal Noise Ratio (PSNR). It is a logarithmic scaling of the normalized inverse of the mean square error: Let $d \in \mathbb{N}$ and S be a finite set of pixels. Then the PSNR value rating the distance of two images $x, y \in \{1, \dots, d\}^S$ is given by

$$\mathrm{PSNR}(x, y) = 10 \cdot \log_{10} \left(\frac{|S| \cdot d^2}{\|x - y\|_2^2} \right).$$

We use this function to rate segmentations because

- it is frequently used in image processing,
- it includes the same distance measure that has been used in the functional,
- it is very fast to compute (in fact it is computed as a by-product in Algorithm 4.2).

In this paragraph we present rate-distortion curves for a collection of images. The PSNR of a segmentation is plotted against the number of pieces for a segmentation class $\mathfrak{S}$. A pair (k, ϱ) on such a curve can be interpreted in the following way: The minimization algorithm computes for k the segmentation yielding the maximal PSNR with at most k pieces in the segmentation class $\mathfrak{S}$. Therefore the curve measures the approximation behavior of $\mathfrak{S}$. A comparison of rate distortion curves for different angles therefore allows to assess the impact of the angular resolution of an approximation. The x-axis of the plots can be understood as a resolution parameter of the wedgelet approximation. Low PSNR values correspond to coarse approximation of the image.

Although our focus is not on image compression we start with an interpretation of the curves via compression. The left end of the curve corresponds to low bit rate compression. A steep ascent of the rate distortion curve for a given image points to a good compression potential of wedgelets for this particular image. Similarly the right end of the curve indicates a lower bound for achievable compression rates, compare the curves for images 'sticks' and 'camera' in Figures 6.41 and 6.40. For instance, the PSNR curves for image 'camera' show an immediate rise to a PSNR value of 25 dB for very

small numbers of pieces. Hence the use of wedgelets for very low bitrate approximation of the camera image seems realistic. This is in accordance with the fact that the camera image consists of large uniform areas with smooth boundaries, i.e. locally the camera image looks like a smooth horizon function. Quite the opposite effect can be witnessed in the plot of image 'sticks' which has extremely poor PSNR values even for a substantial number of pieces. This is an indicator of the poor approximation powers of wedgelets with respect to textures. Similarly the presence of textures in image 'barbara' in Figure 6.43 accounts for a lower ascent of the curve in intermediate regions between 5000 and 20000. By contrast, for artifical geometrical examples such as the image 'circles', a very good approximation is already possible for a small number of segments.

The plots reveal that the gain of incorporating wedge splits over mere quad tree approximations already amounts to several decibels of PSNR. The variations due to angular resolution are on a similar scale, excepting the extremal example 'circles'. Increasing the number of angles has a limited effect. In general one expects an increase of angular resolution to result in a better approximation behavior. This intuition is correct whenever there is an inclusion relation between the respective sets of angles. If this is not the case the opposite can occur. To name an example, for high resolution six angles can be worse than only four. The reason is a discretization effect; for four angles, the set of high resolution wedgelets provides a better fit of image structures. For coarse resolution this is in general not the case, compare the extract in Figure 6.42 for image 'Obelix'.

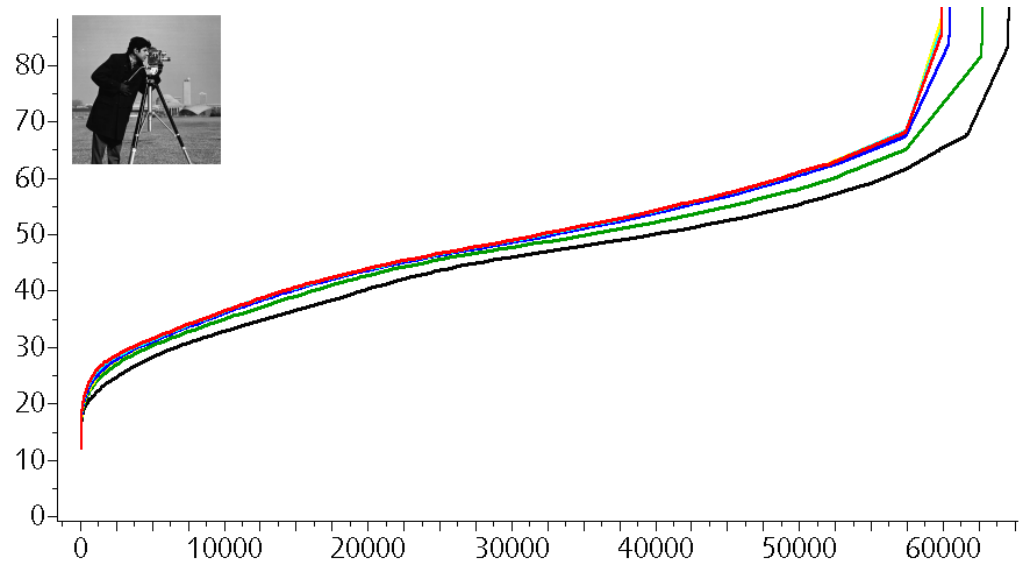


Fig. 6.40: Image 'camera', 256×256. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, red 512.

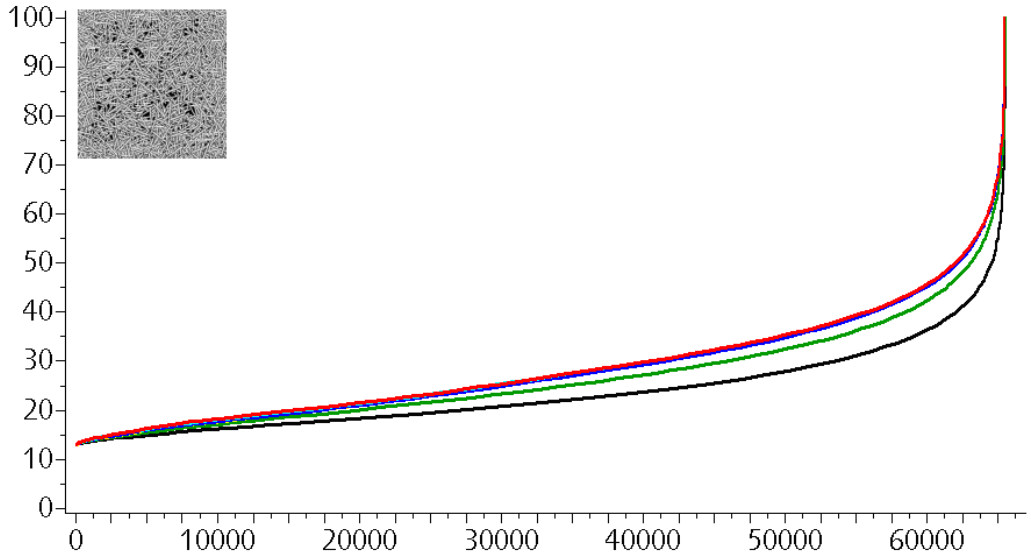


Fig. 6.41: Image 'sticks', 256×256. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, red 512.

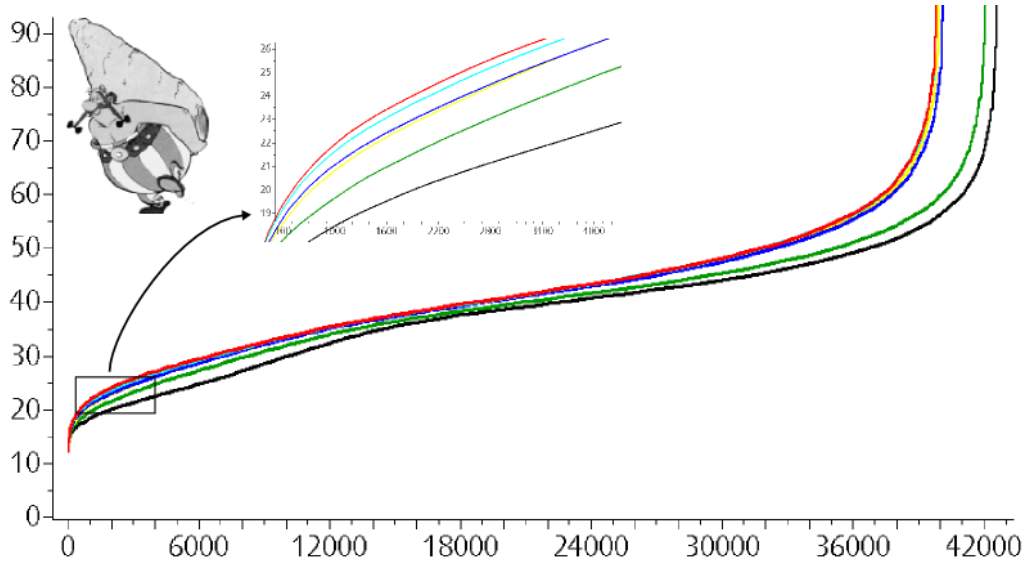


Fig. 6.42: Image 'obelix', 253 × 341. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, red 512.

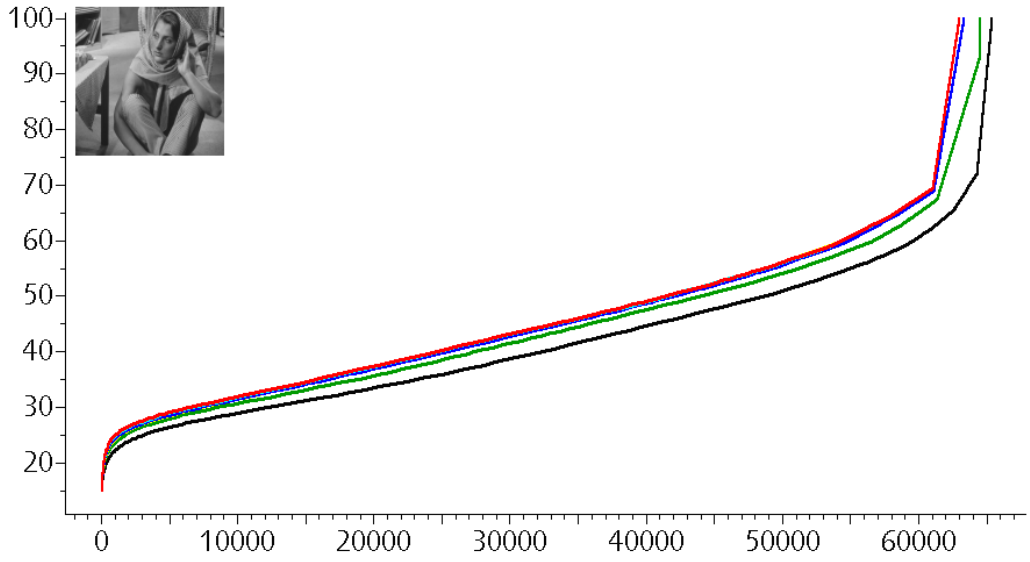


Fig. 6.43: Image 'barbara', 256 × 256. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, red 512.

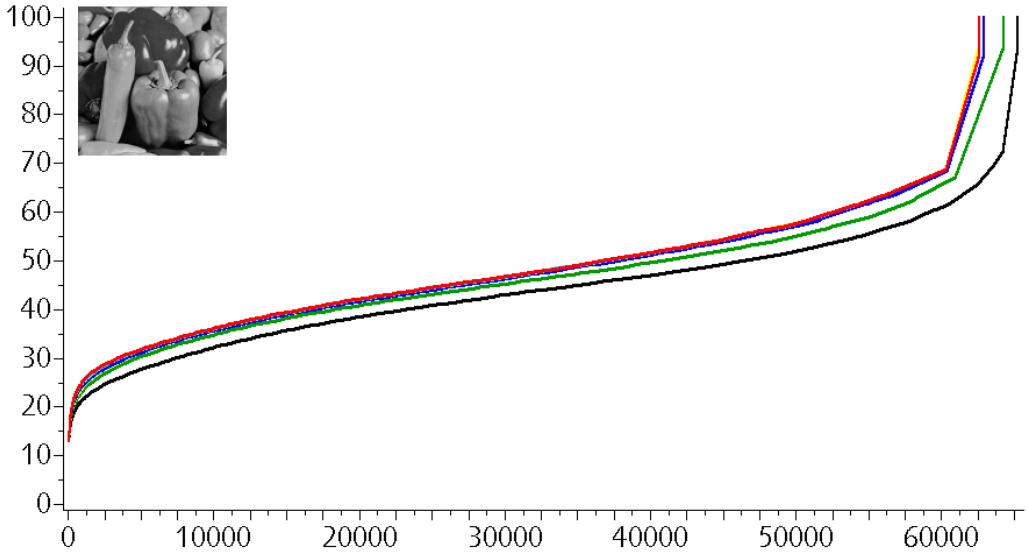


Fig. 6.44: Image 'peppers', 256 × 256. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, red 512.

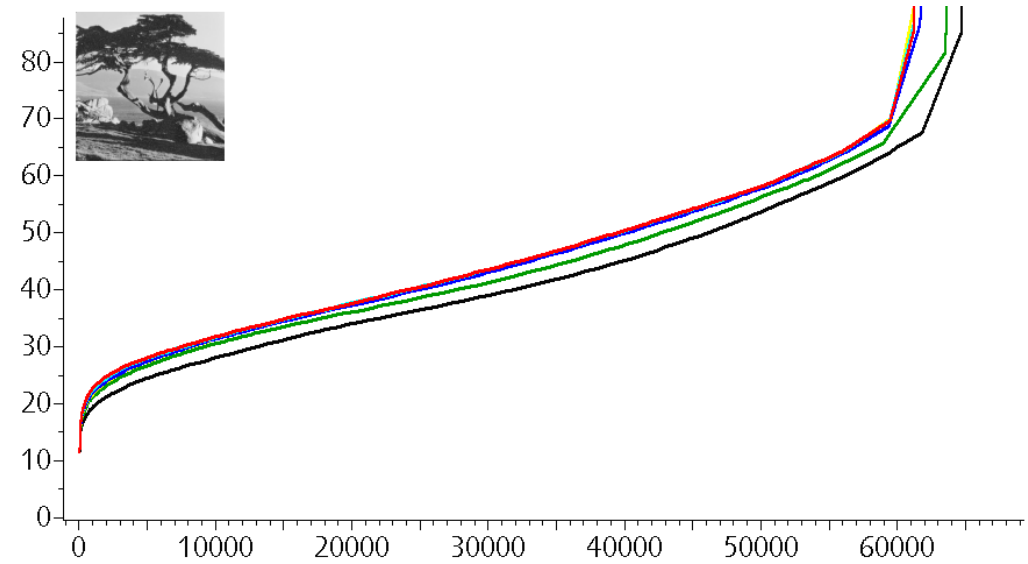


Fig. 6.45: Image 'tree', 256 × 256. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, red 512.

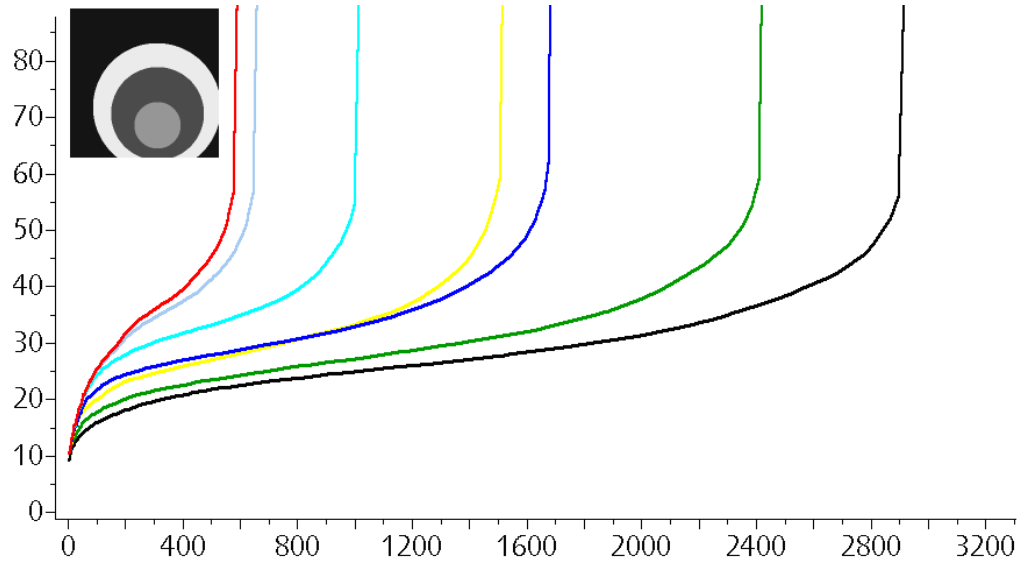


Fig. 6.46: Image 'circles', 256 × 256. PSNR against number of segments for different number of angles: black 0, green 2, yellow 4, blue 6, turquoise 32, lightblue 256, red 512.

Adaptive Angles

Now we motivate the adaptive choice for the set of angles that can be optionally switched on in our software. Let $l \in \mathbb{N}$ denote the number of angles and let the set of angles be given by

$$\Delta_l := \left\{ -\frac{\pi}{2}, -\frac{\pi}{2} + \frac{\pi}{l}, \ldots, \frac{\pi}{2} - \frac{\pi}{l} \right\}.$$

Consider an image domain $S = \{1, \ldots, n\}^2$, $n \in \mathbb{N}$. By Corollary 3.2.18 the number of linear partitions in a grid is bounded from above by n^4. We will not take this number for an estimate of the needed lines but rather consider the lines that connect pixels located on the boundary of S. In fact we will parameterize wedge splits by pairs of points on the boundary and angles. This is motivated by the continuous model where each wedge can be described either by two points on the boundary or by one point on the boundary plus an angle. In the discrete case we are faced with the ambiguities arising from the varying pixelizations of straight lines, compare Paragraph 3.2.2. Nevertheless it is reasonable to assume that the same geometric variety can be generated by using the latter approach, where the number of angles in a given dyadic square scales with the number of boundary points. This reasoning suggests taking $O(2^j)$ angles for dyadic intervals of size 2^j. The PSNR plots below

provide evidence that this procedure yields the same results as taking the full set of n angles in every dyadic scale, for an image of size $n \times n$.

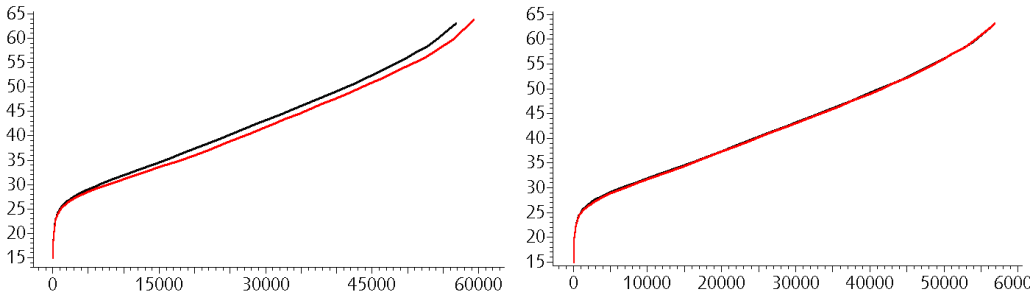


Fig. 6.47: PSNR against number of segments for a wedge model (black) and an adaptive wedge model (red) for image 'barbara', size 256×256, left: 128 angles, right: 256 angles.

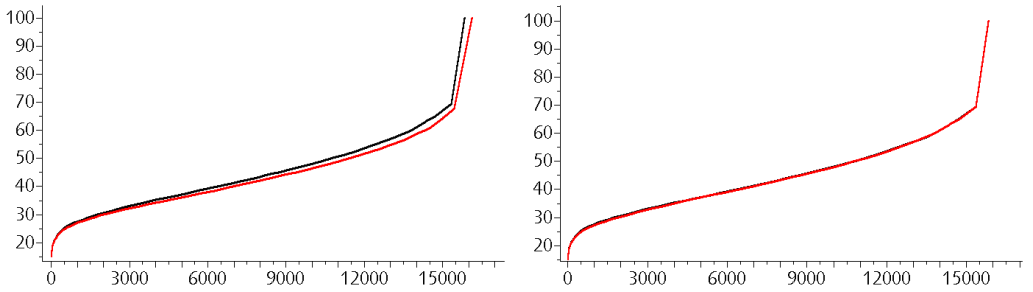


Fig. 6.48: PSNR against number of segments for a wedge model (black) and an adaptive wedge model (red) for image 'barbara', size 128×128, left: 64 angles, right: 128 angles.

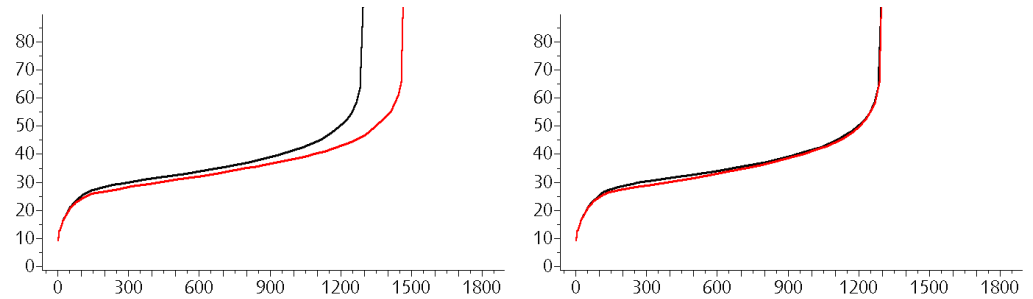


Fig. 6.49: PSNR against number of segments for a wedge model (black) and an adaptive wedge model (red) for image 'circles', size 128×128, left: 64 angles, right: 128 angles.

6.3.4 Horizon Functions

Recall from Theorem 5.3.23 that for a certain class of horizon functions we derived a rate of convergence. Remark 5.3.24 provides a subclass of horizons for which the aforementioned theorem can be applied. In particular for the class of wedge segmentations $\mathfrak{S}$ over $[0,1)^2$ it holds that

$$\inf\{\|g-f\|_2^2 : (\mathcal{P},g) \in \mathfrak{S}, |\mathcal{P}| \leq k\} = O(k^{-2}),$$

if f is the horizon function corresponding to the horizon $x \mapsto x^2$. We have created a horizon function from this horizon for various resolutions and have plotted $\log \|f-\hat{f}\|_2^2$ against number of segments of the associated wedge segmentation, see Figure 6.50. One expects a slope of 2 for this curve at high resolution scales. This cannot be validated with this experiment. A reason is definitely the discretization effect when dealing with too small segments. For a high resolution scale the algorithm approximates the signal much better than in a continuous setting. Even for bigger image sizes the expected slope could not be reasonably extracted from the plot. In the derivation of the rate of the convergence in Donoho (1999), a number of $O(2^j)$ angles is required for squares of size 2^{-j}, i.e. the smaller the rectangles is, the higher the angular resolution is assumed to be. For discrete images, this is obviously not a reasonable assumption and is not in line with the statements of Paragraph 6.3.3.

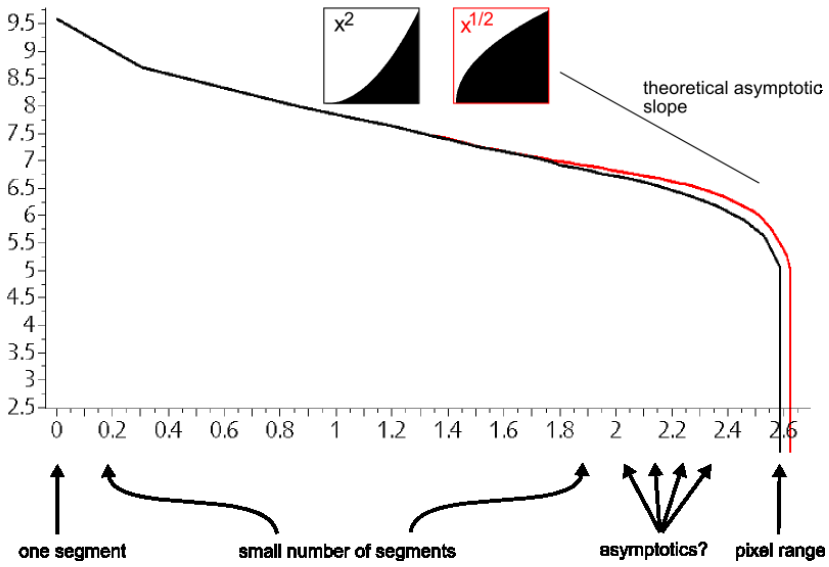


Fig. 6.50: $\log \|f-\hat{f}\|_2^2$ against log number of segments for images horizon $x \mapsto x^2$ (black) and $x \mapsto x^{1/2}$ (red), both 512×512.

6.4 Denoising Experiments

In this section wedge segmentations are investigated with respect to denoising. In Paragraph 6.4.1 some experiments with a collection of noisy images are presented. It will turn out that the presence of noise can be traced in the plot of number of pieces against the parameter γ of a segmentation. In Paragraph 6.4.2 simulations are performed that indicate that the variance of the noise can be estimated from these curves for a large class of distributions.

6.4.1 The Effect

In this paragraph some experiments with noisy images are presented. In the sequel the symbol f will stand for image data, and g will stand for a noisy version of the same image. The symbols $\hat{f}_\gamma$ and $\hat{g}_\gamma$ will stand for segmentations of f and g, respectively, with parameter $\gamma \geq 0$. The following observation can be made frequently: When wedge segmentations of noisy image data g are displayed for different values of γ increasing on a logarithmic scale, then there is a relatively small range of (logarithmic) values of γ where the noise on the resulting image $\hat{g}_\gamma$ suddenly vanishes over the whole image domain, see Figure 6.51. It is desirable to have a reasonable estimator of this breakpoint of regularity.

This resolution parameter of the noise is clearly visible as a minimum on plots displaying the distance $\|f-\hat{g}_\gamma\|_2^2$ of the segmented noisy image to the original image against the logarithm of the parameter γ. It can also be detected when plotting $\|\hat{f}_\gamma - \hat{g}_\gamma\|_2^2$ against $\log\gamma$ roughly at the steep descent of the curve, see Figure 6.53. This observation is not particularly surprising. Moreover, it cannot be used in real applications since the original image is unknown.

Interestingly the plot of the number of pieces of a segmentation of noisy data against the logarithm of the corresponding parameter γ reveals a similar behavior as $\|\hat{f}_\gamma - \hat{g}_\gamma\|_2^2$ against $\log\gamma$ for small values of γ. In Figures 6.53-6.59 the resolution parameter can be roughly estimated by means of these curves. For a priori smooth images the difference between the plots for noisy images and those for the original image is visible more clearly than for images with many details on the high resolution scale, compare Figure 6.57 and 6.55. Moreover, the effect gets stronger for increasing noise, see Figures 6.57 and 6.59. The last two figures do also suggest that for varying noise the curve is translated horizontally. In the next paragraph, we will experimentally investigate this dependency of the noise in more detail.

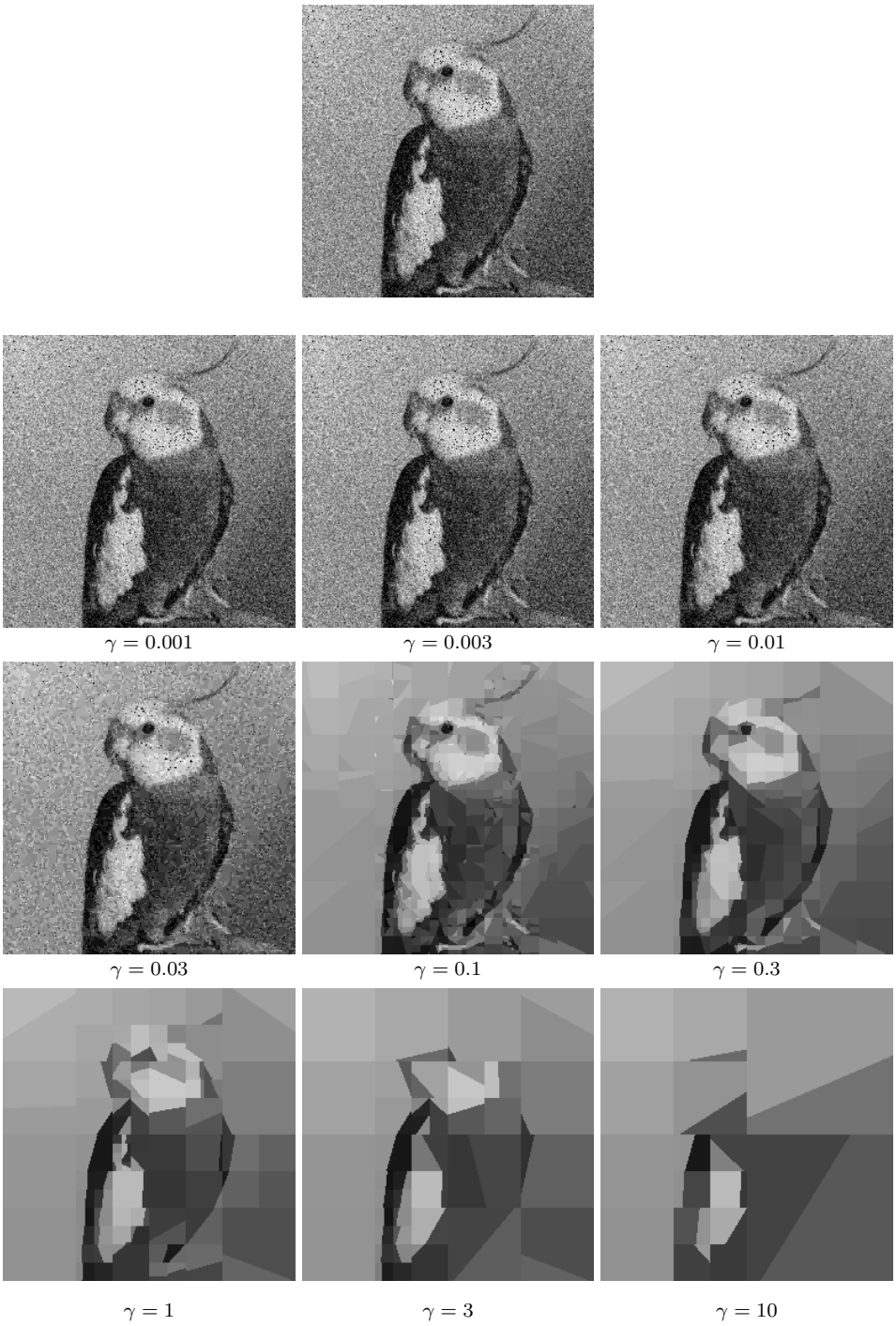


Fig. 6.51: Wedge segmentations of noisy image 'bird'. Original image distorted by Gaussian noise with variance 1000 (top). Dyadic wedge segmentations with 512 adaptive angles and constant regression for different parameter values of γ on a logarithmic scale. Observe the sudden vanishing of noise roughly at $\gamma = 0.03$.

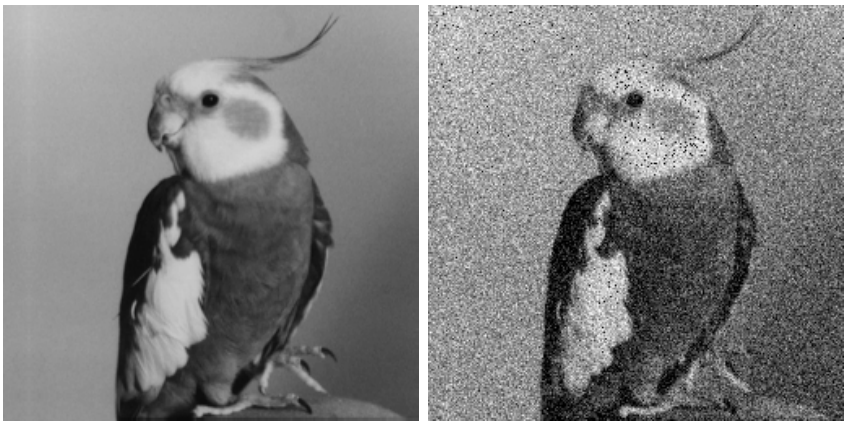


Fig. 6.52: Image 'bird', original f (left) and distorted version g, Gaussian noise with variance 1000 (right).

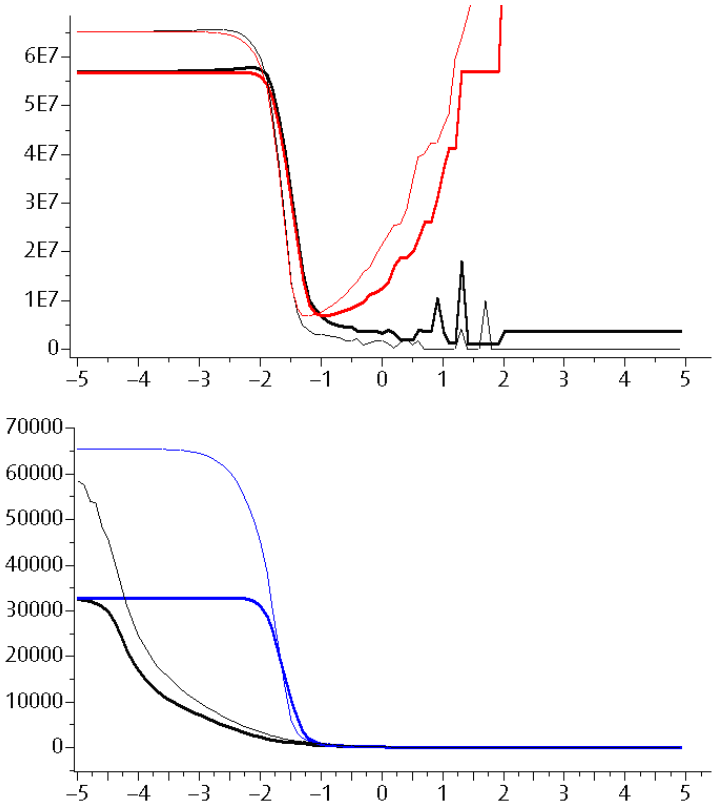


Fig. 6.53: Plots to image 'bird'. Thin lines: dyadic square model, thick lines: dyadic wedge model with 512 adaptive angles. Upper plot: $\|\hat{f}_\gamma - \hat{g}_\gamma\|_2^2$ (black) and $\|\hat{g}_\gamma - f\|_2^2$ (red) versus $\log\gamma$. Lower plot: number of pieces of $\hat{g}_\gamma$ (blue) and of $\hat{f}_\gamma$ (black) versus $\log\gamma$.

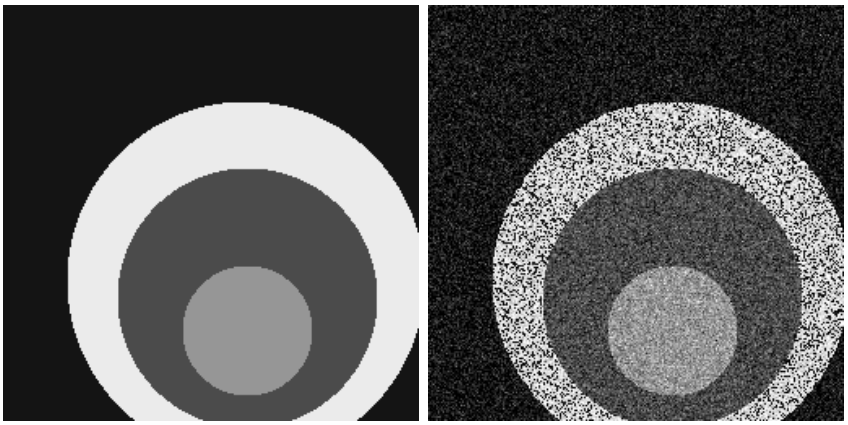


Fig. 6.54: Image 'circles', original f (left) and distorted version g, Gaussian noise with variance 1000 (right).

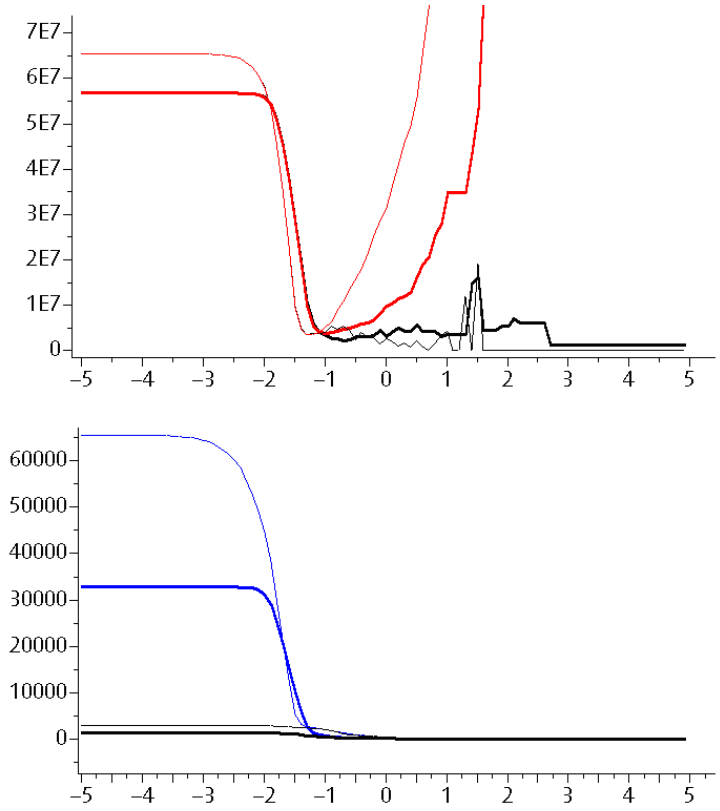


Fig. 6.55: Plots to image 'circles'. Thin lines: dyadic square model, thick lines: dyadic wedge model with 512 adaptive angles. Upper plot: $\|\hat{f}_\gamma - \hat{g}_\gamma\|_2^2$ (black) and $\|\hat{g}_\gamma - f\|_2^2$ (red) versus $\log \gamma$. Lower plot: number of pieces of $\hat{g}_\gamma$ (blue) and of $\hat{f}_\gamma$ (black) versus $\log \gamma$.

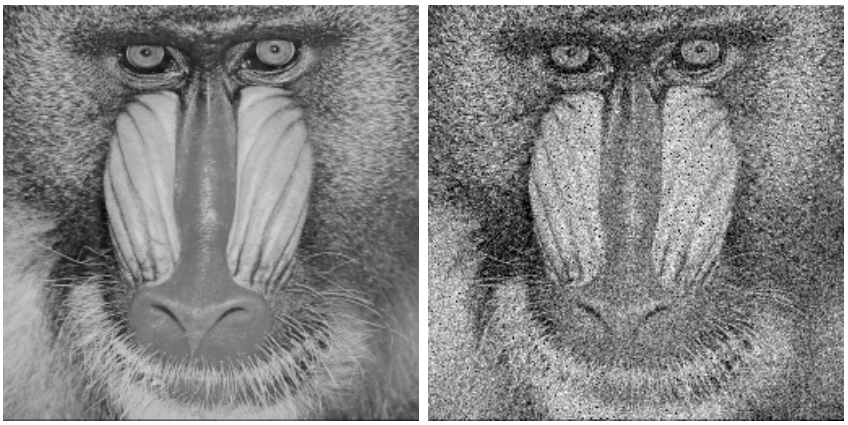


Fig. 6.56: Image 'mandrill', original f (left) and distorted version g, Gaussian noise with variance 1000 (right).

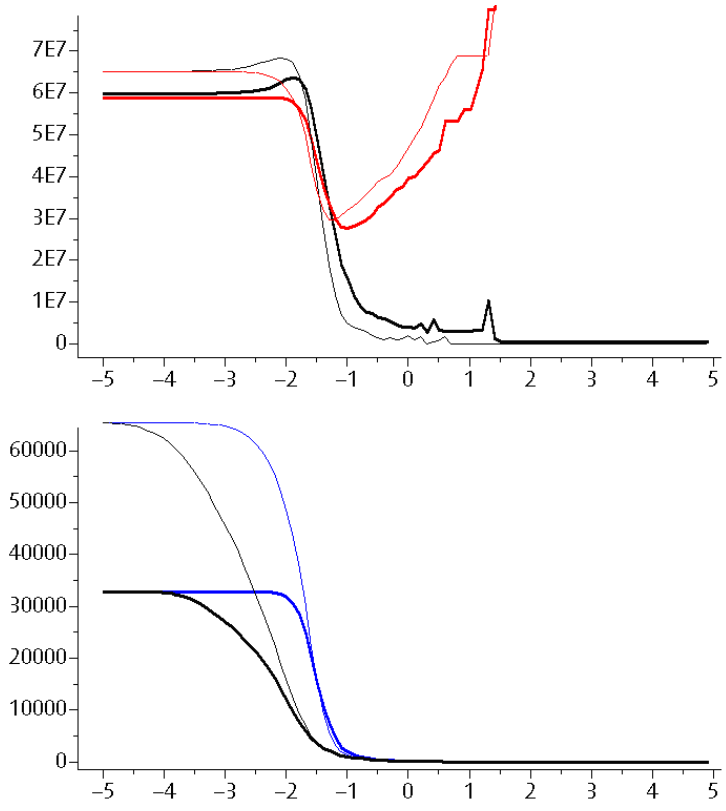


Fig. 6.57: Plots to image 'mandrill'. Thin lines: dyadic square model, thick lines: dyadic wedge model with 512 adaptive angles. Upper plot: $\|\hat{f}_\gamma - \hat{g}_\gamma\|_2^2$ (black) and $\|\hat{g}_\gamma - f\|_2^2$ (red) versus $\log\gamma$. Lower plot: number of pieces of $\hat{g}_\gamma$ (blue) and of $\hat{f}_\gamma$ (black) versus $\log\gamma$.

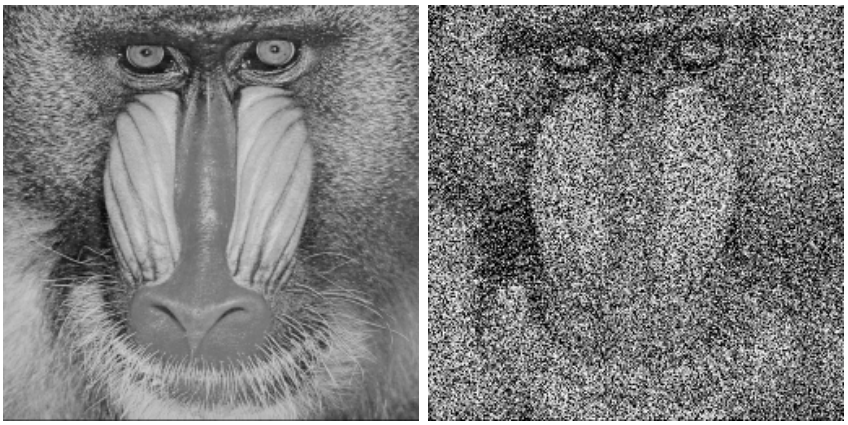


Fig. 6.58: Image 'mandrill', original f (left) and distorted version g, Gaussian noise with variance 5000 (right).

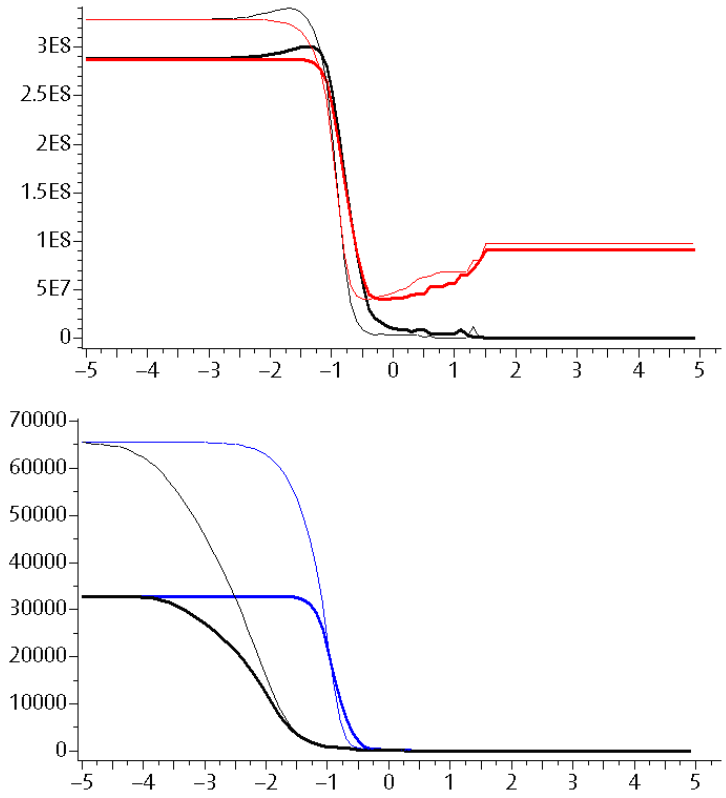


Fig. 6.59: Plots to image 'mandrill'. Thin lines: dyadic square model, thick lines: dyadic wedge model with 512 adaptive angles. Upper plot: $\|\hat{f}_\gamma - \hat{g}_\gamma\|_2^2$ (black) and $\|\hat{g}_\gamma - f\|_2^2$ (red) versus $\log \gamma$. Lower plot: number of pieces of $\hat{g}_\gamma$ (blue) and of $\hat{f}_\gamma$ (black) versus $\log \gamma$.

6.4.2 Detecting the Noise Level

In this paragraph curves displaying the log of number of segments against $\log \gamma$ are investigated. A comparison of these plots for the original data, the noisy image and a noisy flat distorted with the same distribution show that the curves for noisy data are in fact given as a superposition of the ones for the noise and the original data. They roughly consist of two components. One for the noise in the high resolution part and one for the signal in the lower resolution part. This division can be made out the clearer the smoother the original image is.

The plots below display the number of segments against $\log \gamma$ for segmentations on the original image, a noisy version and a noisy flat, both noised with the same distribution. Thin lines correspond to a dyadic square model while thick lines correspond to a wedge adaptive model with 512 angles. All images are of size 256×256.

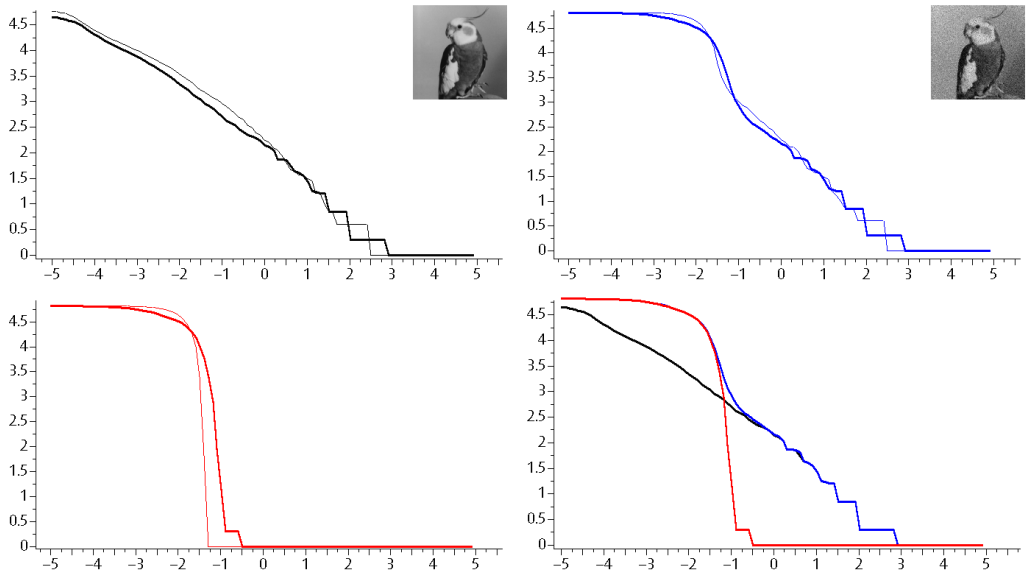


Fig. 6.60: Plots for image 'bird'. Log number of segments against $\log \gamma$ for segmentations on the original image, a noisy version and a noisy flat, variance 1000. Lower right: Overlay of the curves.

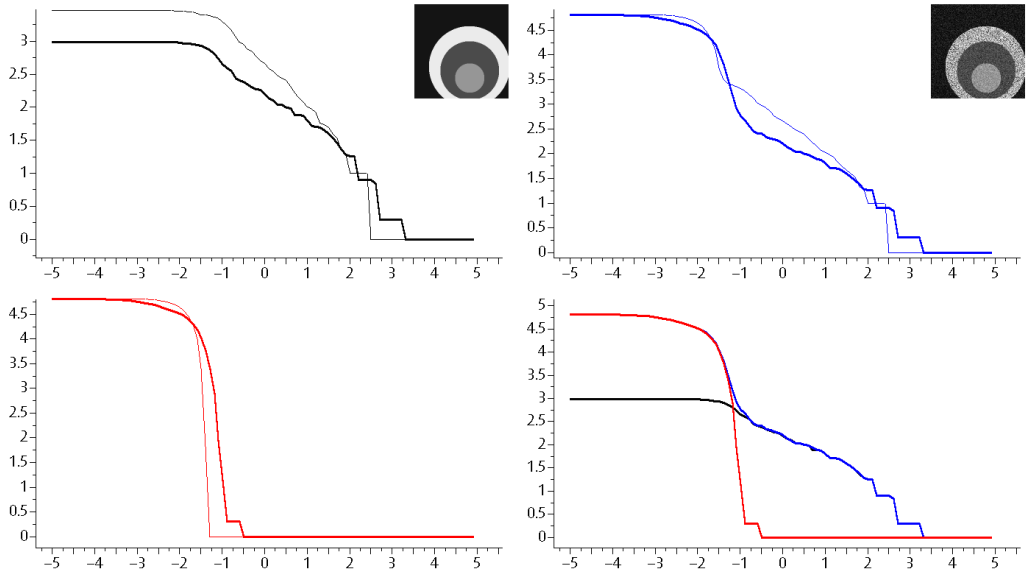


Fig. 6.61: Plots for image 'circles'. Log number of segments against $\log \gamma$ for segmentations on the original image, a noisy version and a noisy flat, variance 1000. Lower right: Overlay of the curves.

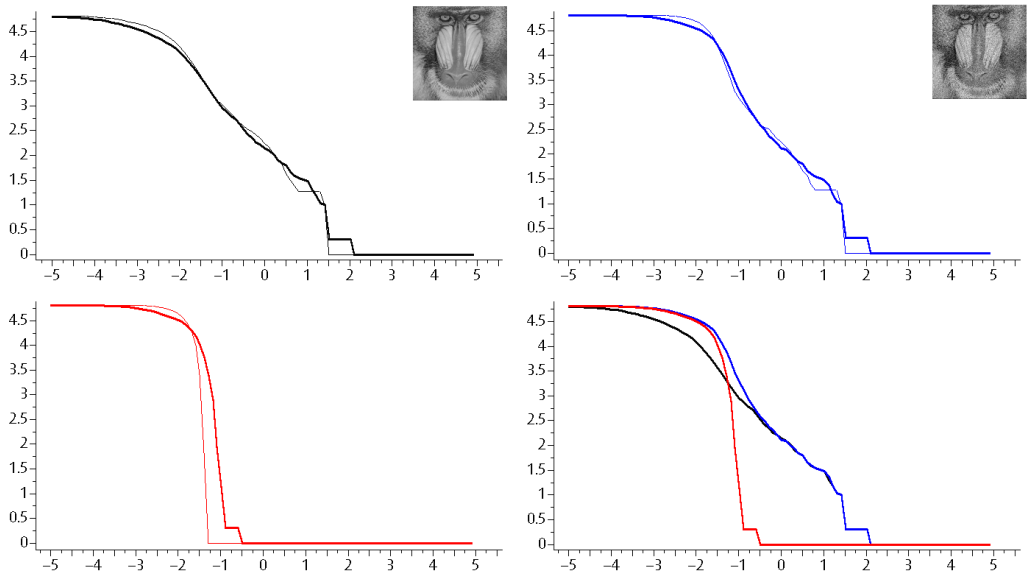


Fig. 6.62: Plots for image 'mandrill'. Log number of segments against $\log \gamma$ for segmentations on the original image, a noisy version and a noisy flat, variance 1000. Lower right: Overlay of the curves.

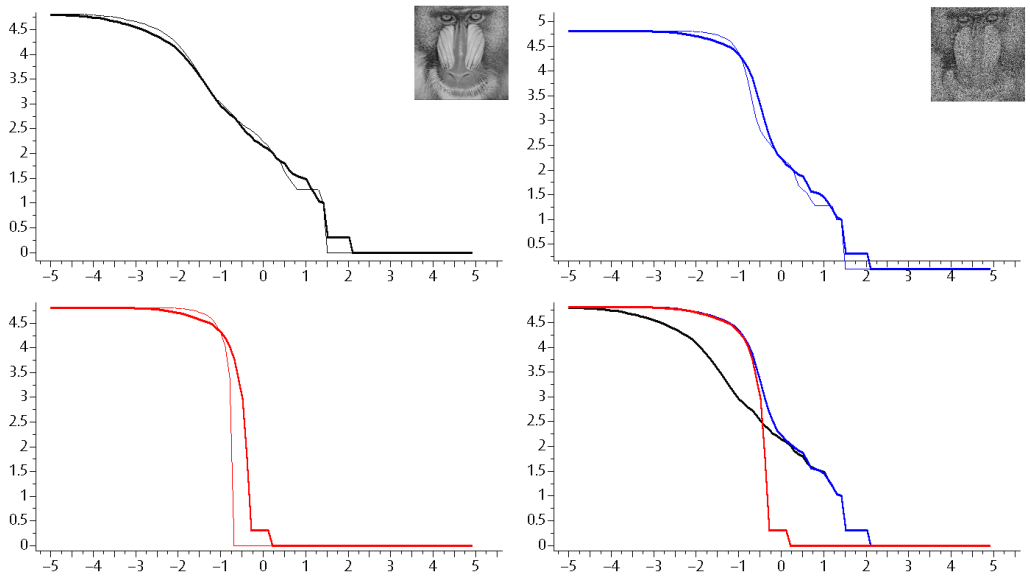


Fig. 6.63: Plots for image 'mandrill'. Log number of segments against $\log \gamma$ for segmentations on the original image, a noisy version and a noisy flat, variance 1000. Lower right: Overlay of the curves.

A comparison of the Figures 6.62 and 6.63 suggests that a change of the variance of the noise yields a horizontal translation of the noise curve. Additionally, if the variance remains unchanged this curve seems to be independent of the underlying noise for a large class of distributions. We have simulated a noisy flat for a normal distribution, a uniform distribution and a Laplace (double exponential) distribution, all of them with different variances. Figure 6.64 shows that the plots are nearly independent of the underlying distributions, which provides a hint to the robustness of corresponding estimators of the variance or the noise resolution parameter.

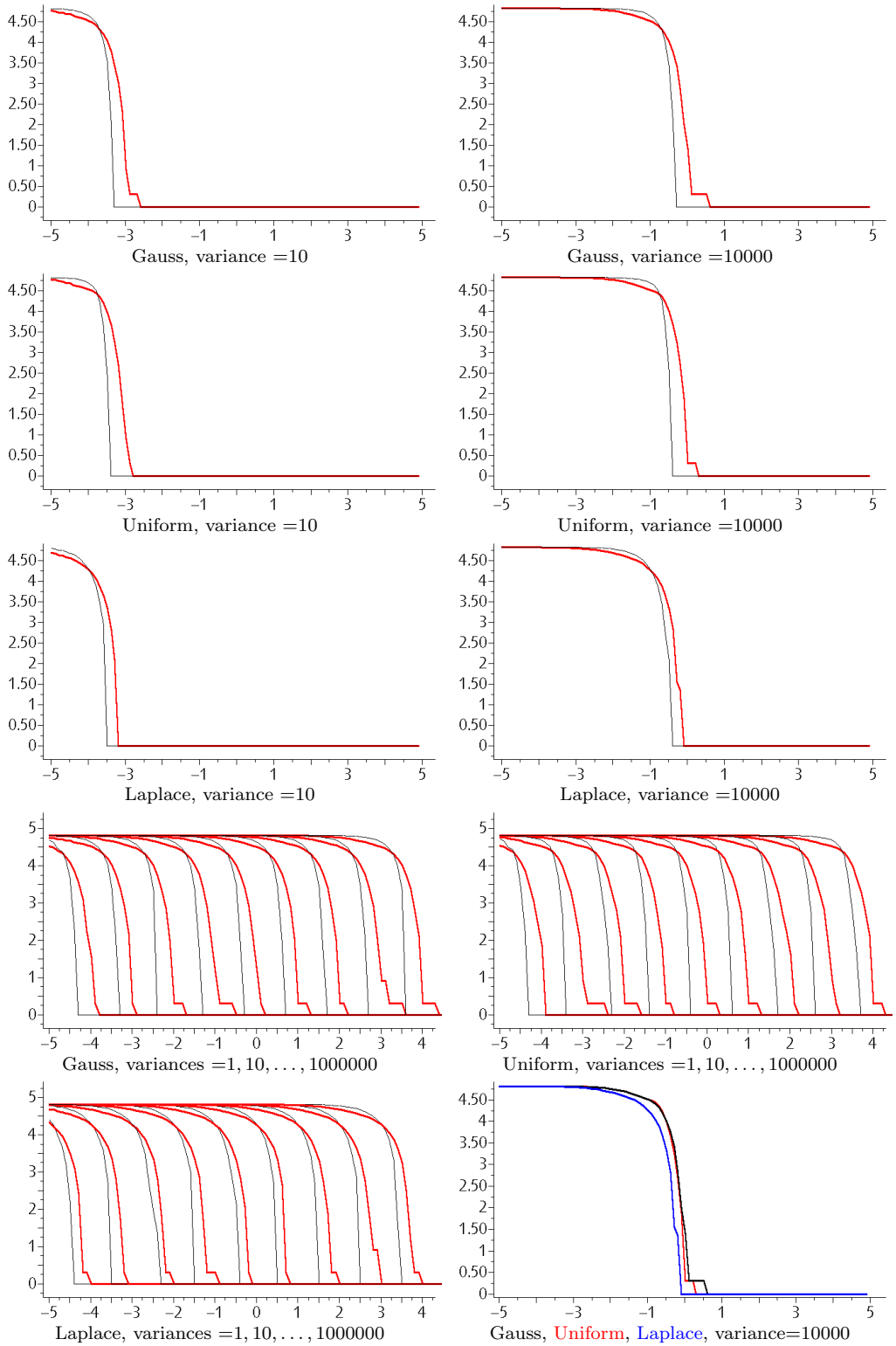


Fig. 6.64: Plots for a noisy flat. Thin lines: dyadic square model, thick lines: dyadic wedge model with 512 adaptive angles. Displayed are *log* number of segments agains $\log \gamma$.

Figure 6.64 also suggests that the translation of the curves is log linear dependent of log of the variance. In the following plots we have displayed the left most point on the noise curves with $|\hat{\mathcal{P}}_\gamma| \leq 10$ as an estimator of the horizontal translation of the curve, and obtained the log log linear dependency displayed in Figure 6.65. This can be used to estimate the variance in a noisy image. Experiments where the curves of a noisy flat are adjusted to the curves of a noisy image in an interactive environment, showed that indeed the variance of the added noise can be determined quite accurately. An automatic determination should thus not be very difficult.

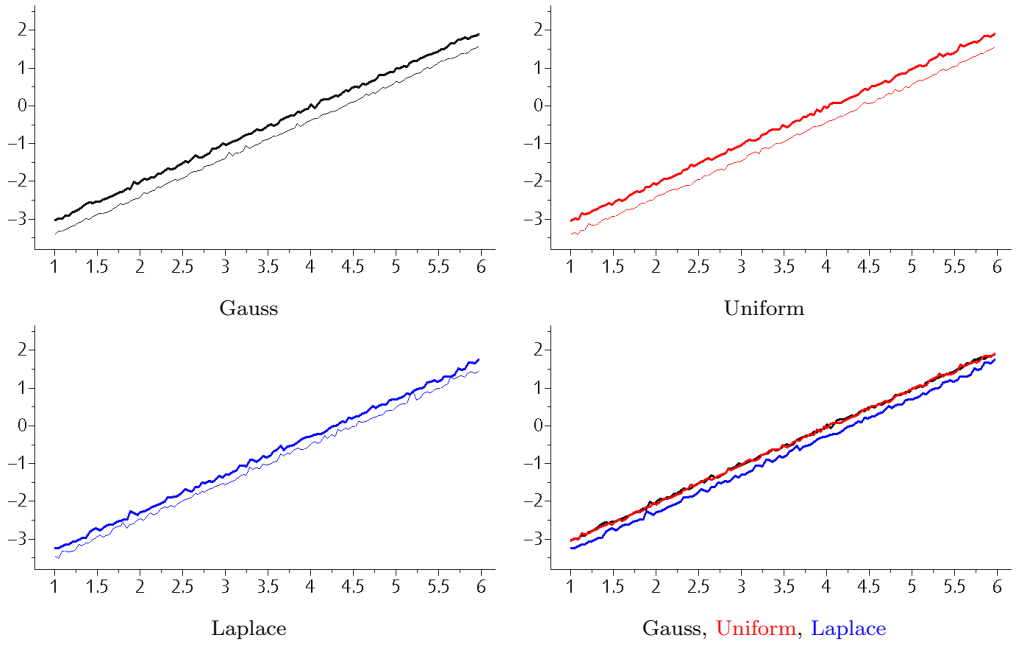


Fig. 6.65: Plots for a noisy flat. Thin lines: dyadic square model, thick lines: dyadic wedge model with 512 adaptive angles. Displayed are $\log(\inf\{\gamma > 0 : |\hat{\mathcal{P}}_\gamma| \leq 10\})$ against log variance of the noise.

6.5 Synopsis

This chapter includes experiments that have been performed with the algorithms presented in this thesis. All algorithms have been implemented in an interactive environment and will be made available online. A runtime analysis experimentally verified the statements about complexity in Theorem 4.3.1 and made clear that dyadic wedge segmentations can be computed quickly for reasonable image dimensions. Hierarchic segmentations can be computed

in bearable time only with certain restrictions and are therefore rather of theoretical relevance.

We have strictly separated the local regression part from the minimization and reconstruction part of the algorithms, both in theory and in the implementation. This yielded the following striking feature: Once some precomputation steps have been performed, the Potts functional can be minimized more or less immediately for each resolution parameter γ for reasonable image sizes. A consequence is that we consider the result of the algorithm to be rather the set of images for all parameters γ than only a single minimization result. This gave rise to experiments that take advantage of this multiresolution approach.

Plots of the PSNR values against the number of segments of a segmentation for different natural and artifical images, reveal features of wedge segmentations with respect to potential compression rates. It turned out that incorporating wedge splits gains already some decibel in comparison to mere quad tree segmentations. For natural images an increase of the number of angles beyond an amount of 32 does not radically improve the approximation in terms of PSNR. For geometric images this limit can be identified at about an amount of $256 - 512$ angles. For even faster performance of the precomputation steps, we proposed an adaptive choice of the angular resolution. There the number of angles is chosen to be $O(2^j)$ for dyadic intervals with side lengths 2^j. Comparisons of this adaptive scheme with taking the full angle set for all resolutions reveal a nearly unchanged PSNR curve, if an initial amount of l angles for images with size $l \times l$ is taken.

In experiments with noisy images an interesting effect can be observed. When wedge segmentations are displayed for values of γ on a logarithmically increasing scale, then there is a small range of γ where the noise suddenly vanishes. This breakpoint of γ can be identified on plots displaying the (log) number of segments against $\log \gamma$. Simulations of noisy images and a noisy flat for different distributions and different variances have revealed that the noise can be identified by a certain slope in the corresponding curves. This suggests a method for estimation of both the variance and the noise resolution parameter for the purpose of denoising images.

Discussion and Outlook

In this discussion, we display desirable future work in *italics*.

We have considered minimization of functionals of the form

$$H_\gamma : \mathfrak{S} \to \mathbb{R}, (\mathcal{P}, f_\mathcal{P}) \mapsto \gamma \cdot |\mathcal{P}| + \|f_\mathcal{P} - f\|_2^2, \gamma \geq 0,$$

on the set of wedge segmentations $\mathfrak{S}$ over image domain S.

For these particular functionals and for two classes of wedge segmentations we have presented efficient minimization schemes. These are based on an efficient recursion scheme over partitions and a fast computation of the local regressions on each fragment of a partition. We have provided efficient algorithms performing the local regression for any finite dimensional function space over polygonal domain that require a precomputation of certain matrices for all directions within the respective polygon. Thereby we have presented an algorithm for efficient minimization of Potts functionals over wedge segmentations with arbitrarily adjustable angular resolution.

As already mentioned in Chapter 1, efficient minimization algorithms are possible for a wider class of functionals by means of the reduction principle. If it holds, then a minimization of the functional can be performed by separate computation of the local minimization for each fragment and the minimization over the set of partitions, i.e. the recursion schemes for dyadic and hierarchic partitions remain valid. For example, the algorithms can be (and have already been) applied to a functional of the form $(\mathcal{P}, f_\mathcal{P}) \mapsto \gamma \cdot \sum_{r \in \mathcal{P}} \text{Edge Length}(r) + \|f_\mathcal{P} - f\|_2^2$ with minor modifications. (In this case the immediate accessibility of minimizers for each γ after some precomputation steps for the dyadic model would be lost.) However, a modification of the data term $\|f_\mathcal{P} - f\|_2^2$ to other distances, such as the more robust L^1 distance, does not allow for an easy generalization of the results of this thesis concerning the local regression part. The problem is, that the computation of moments such as the mean over regions of the image must then be replaced by nonlinear functions of the data, such as, for instance, segmentwise medians. *It would be of high interest to find similar fast computation methods for other distances than L^2. Willett and Nowak (2003) suggest a modification of the functional that is intended as an approximate solution to this problem.*

A straightforward extension of the minimization algorithms to problems, where an explicit interaction between different fragments of a partitions is of importance, is not possible, since this would violate the reduction principle. *However, the statements about efficient regression over polygonal domain are*

self contained and might be of use for the solution of other minimization problems. There, for instance, we think also on more complex partition schemes.

Since the local regression part of the minimization algorithms has already been formulated and implemented for basically any class of function spaces, namely finite dimensional function spaces, *an analysis using completely different function spaces would be interesting. A use for better resolution of patterns is imaginable.*

Completely new geometries could be adopted. The summation trick used for the computation of moments in polygons works in similar form for lines, curves and other shapes.

We have investigated segmentation classes with respect to consistency. The requirement on the noise to be subgaussian is quite mild and it would not make much sense to invest additional effort into generalizations. We showed that, if a certain approximation rate is given on the continuous side and if the discretization error is bounded in a given way, then consistency and even a rate of convergence hold for a wide class of segmentations. We have applied this result to dyadic and hierarchic wedge segmentations. Two scenarios from literature, piecewise polynomial approximation of smooth functions and constant approximations of horizon functions, have also be investigated. We expect, that the results hold for more general regression models. For piecewise polynomial approximations we obtained consistency for the hierarchic wedge segmentations and additionally a rate of convergence for dyadic wedge segmentations and sufficiently smooth original image. For constant approximations, we address the problem of discretization and approximation separately and formulate results for horizon functions. In this special case for hierarchic segmentations we have proved a convergence rate that is superior to that of dyadic segmentations. In the one-dimensional case, the nonlinear approximation spaces of functions approached at a given rate are well known for a great variety of function spaces, compare DeVore (1998). For wedgelets and derivatives, in the two-dimensional set up, only few, partial results exist, e.g. for horizon functions with Hölder-regular boundaries, compare Donoho (1999), pp. 866-873. Since horizon functions are one dimensional in nature, these results are usually based on one-dimensional approximation results applied to the boundary. *It should be stressed that the existing results do not fully exhaust the adaptivity of dyadic partitions. A related open problem is the development of sharp results, such as the characterization of function spaces by their approximation behavior.*

In applications, the parameter γ has to be chosen in an adequate way. The asymptotics derived in Chapter 5 do not provide sufficient information in this respect, since only the optimal asymptotics for γ were determined. Even for the case of one-dimensional signals, this problem is not solved in a satisfactory manner. The non-asymptotic estimate derived in Paragraph 5.2.2 gives additional information that could be used for the choice of the parameter γ.

All algorithms presented in this thesis have been implemented in an interactive environment and will be made available online. A runtime analysis experimentally verified the statements about runtime and memory consumption of the algorithms. By a strict separation of the local regression part from the minimization and reconstruction part of the algorithms, both in theory and in the implementation, we obtained, that, once some precomputation steps have been performed, the Potts functional can be minimized in real time for each resolution parameter γ. This gave rise to experiments that take advantage of a multiresolution approach.

Plots of the PSNR value against the number of segments of a segmentations for different natural and artifical images have given hints that wedge segmentations are well suited for compression. This was not in the focus of this thesis. *Thus, a detailed analysis of wedge segmentations with respect to compression would clearly be of interest. For this purpose, an adaption of the functional, incorporating coding cost, will be instrumental.*

It turned out, that for natural images an increase of the number of angles beyond an amount of 32 does not radically improve the approximation in terms of PSNR. For geometric images this limit could be identified at about an amount of $256-512$ angles. For even faster performance of the precomputation steps, we proposed an adaptive choice of the angular resolution leading to further speed-up. Comparisons of this adaptive scheme with taking the full angle set for all resolutions revealed a nearly unchanged PSNR curve, if an initial amount of l angles for images with size $l \times l$ is taken.

Experiments showed a great potential of the methods provided here for the purpose of image denoising. When wedge segmentations are displayed for values of γ on a logarithmically increasing scale, then there is a small range of γ where the noise suddenly vanishes. This breakpoint of γ can be identified on plots displaying the (log) number of segments against $\log \gamma$. Simulations of noisy images and a noisy flat for different distributions and different variances have revealed that the noise can be identified by a certain slope in the corresponding curves. *This suggests a method for estimation of both the vari-*

ance and the noise resolution parameter for the purpose of denoising images. Since it is both easy to implement and important for practical application, further studies of this method will be done in near future. Beside that, theoretical results are desirable.

The development of multiscale techniques employing the whole range of minimizers seems to be a rather promising and rich field for new research. In particular, the directional information coded in wedgelet approximation should be of use.

Fields of possible applications include segmentation, denoising, compression, image resampling and edge detection of images.

Symbols

Greek

β	universal constant of subgaussian noise	129
Δ	set of angles	103
$\boldsymbol{\delta}$	canonical discretization	20
$\boldsymbol{\delta}^n$	discretization of continuous function to function over S_n	125
$\boldsymbol{\delta}^R(\mathcal{P})$,	discretization of partition $\mathcal{P}$ of $[0,1)^2$ in R	37
$\boldsymbol{\iota}^n$	embedding of a function over S_n to a function over $[0,1)^2$	125
$\Lambda_n(\Delta, R)$	set of simple polygons in R with n vertices and angles Δ	100
$\xi^{(n)}$	matrix of independent random variables	124
$\mathit{\Pi}_{\mathcal{P}}$	orthogonal sum of projections $\bigoplus_{r\in\mathcal{P}} \mathcal{F}_r$	134
$\mathit{\Pi}_{\mathcal{F}_r}$	projection onto function space $\mathcal{F}_r$, $r \in \mathcal{R}$	28
$\varrho_i^{\blacktriangledown}((0,1))$	subgraph of the edge ϱ_i of polygon ϱ	96
$\varrho_i^{\triangledown}((0,1))$	strict subgraph of the edge ϱ_i of polygon ϱ	96

Roman

$\mathrm{bd}(n)(r)$	discrete boundary of $r \subset [0,1)^2$	152
C^m	Space of m times continuously differentiable functions.	147
dir_φ	horizontal direction functions for homeomorphism φ	92
$\mathfrak{D}$	mapping from adjacent points p, q to a linear dichotomy	80
D	dichotomy	78
$\mathcal{F}$	class of admissible functions	22
$f^{(n)}$	averaged version of $f : [0,1)^2 \to \mathbb{R}$ over $n \times n$ grid	124

$f_{\mathcal{P}}$	family of functions $f_r, r \in \mathcal{P}$	23
$\bar{H}_\gamma$	Potts functional on continuous domain $[0,1)^2$	127
$\tilde{H}_\gamma^{(n)}$	discrete Potts functional embedded into $[0,1)^2$	127
H_γ	Potts functional to parameter γ	28
$H_\gamma^{(n)}$	Potts functional with normalized data term	127
$I(\varphi)$	inside of Jordan curve $\varphi(\mathbb{T})$	86
$I^\alpha(f)$	cumulative sum arrays with angle α to function f	102
$I_s^{(n)}$	$1/n \times 1/n$ cells of $[0,1)^2$, $s \in S_n$	124
$d_{ab}(s)$	signed weighted distance of $s \in \mathbb{R}^2$ to line through $a, b \in \mathbb{R}^2$	78
$L(V_n)$	set of terminal nodes of quad tree (V_n, E_n)	46
$L_{d,\alpha}$	discrete line with intercept d and angle α	73
$\mathrm{map}(A, B)$	set of functions from set A to set B	22
$\mathbb{N}$	set of integers greater or equal zero	19
$N_{\varphi,\psi}(s)$	crossing direction function for two jordan curves φ, ψ	87
$O(\varphi)$	outside of Jordan curve $\varphi(\mathbb{T})$	86
$O(g(n))$	order of $g(n)$	29
$(\mathcal{P}, f_{\mathcal{P}})$	segmentation	22
$\mathcal{P}$, $\mathcal{Q}$	partition	20
$\mathfrak{P}$	partition class	20
$\mathfrak{P}_{\leq k}^{(n)}$	class of hierarchic square partitions with bounded vertical divisions	145
$\mathfrak{W}_{\leq k}^{(n)}$	class of hierarchic wedge partitions with bounded vertical divisions	145
$p^{\blacktriangledown}$	subgraph of point p	96
$p^{\triangledown}$	strict subgraph of point p	96

$\mathcal{R}$	set of admissible fragments	20
$\mathbb{R}$	set of real numbers	19
$r^{(n)}$	discretization of set $r \subset [0,1)^2$	152
$\mathfrak{S}$	segmentation class	22
S	image domain	20
S_n	domain $\{1,\ldots,n\}^2$, $n \in \mathbb{N}$	124
$\mathcal{T}_{x,\alpha}(r)$	division of a rectangle by a line with offset x and angle α into wedges $A_{x,\alpha}(r)$, $B_{x,\alpha}(r)$	69
$\mathbb{T}$	unit circle, $\mathbb{T} = \{(x,y) \in \mathbb{R}^2 : x^2 + y^2 = 1\}$	85
(V_n, E_n)	(quad) tree with nodes V_n and edges E_n, $n \in \mathbb{N}$ number of quad split operations.	45
$\mathcal{V}^{\blacktriangledown}(M)$	points vertically below or in M	74
$\mathcal{V}^{\vartriangle}(M)$	points strictly above M	74
$\mathfrak{W}$	wedge segmentation class	115
$\bar{\mathcal{Y}}_i(\varrho, z), \mathcal{Y}_i(\varrho, z)$	sum of values of z below line ϱ_i of polygon ϱ	96
$\mathbb{Z}$	set of integers	19

Other Symbols

$\mathcal{P} \wedge \mathcal{Q}$	superposition of partitions $\mathcal{P}$ and $\mathcal{Q}$	135
$\lceil x \rceil$	smallest integer greater than or equal to x	101
$\lfloor x \rfloor$	largest integer less than or equal to x	101
$\prec$	half order on $\mathbb{R}^n$, $n \in \mathbb{N}$	27
$A \boxtimes B$	$\{a \times b : a \in A, b \in B\}$	45
$A \ominus B$	symmetrical difference of set A and B	151
∂	neighborhood relation	20

References

M. Abramovitz and I.A. Stegun, editors. *Handbook of Mathematical Functions*. Dover Publications, NewYork, 9 edition, 1972.

D.M. Acketa and J.D. Žunić. On the number of linear partitions of the (m, n)-grid. *Information Processing Letters*, 38:163–168, 1991.

L. Boysen, A. Kempe, V. Liebscher, A.Munk, and O. Wittich. Consistencies and rates of convergence of jump-penalised least squares. manuscript, 2004.

S. Brandt and A. Hutzenthaler. Eine Simulationsbases zur Minimierung von Potts Funktionalen. Projektbericht, Institut für Biomathematik und Biometrie, GSF Forschungszentrum für Umwelt und Gesundheit, München-Neuherberg, 2004.

J. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems J.*, 4:25–30, 1965.

E.J. Candès. Harmonic analysis of neural networks. *Appl. Comput. Harmon. Anal.*, 6:197–218, 1999.

E.J. Candès and D.L. Donoho. Ridgelets: a key to higher-dimensional intermittency? *Phil. Trans. R. Soc. Lond. A.*, 357:2495–2509, 1999.

G. Casella and C.P. Robert. *Monte Carlos Statistical Methods*. Springer Texts in Statitstics. Springer Verlag, Berlin, Heidelberg, New York, 1999.

R.A. DeVore. Nonlinear approximation. *Acta Numerica*, pages 51–150, 1998.

M. N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *http://www.ifp.uiuc.edu/*, to appear.

D.L. Donoho. Wedgelets: Nearly minimax estimation of edges. *Annals of Statistics*, 27(3):859–897, 1999.

D.L. Donoho, A.G. Flesia, and X. Huo. Beamlab. *http://www-stat.stanford.edu/beamlab/, 8.6.2004*, 2004.

James Dugundi. *Topology.* Allyn and Bacon, Inc., Boston, 470 Atlantic Avenue, Boston., 9th edition, 1974.

F. Friedrich. AntsInFields*: A Software Package for Random Field Models and Imaging.* Institute of Biomathematics and Biometry, National Research Center for Environment and Health, Munich, Germany, 2002.

F. Friedrich. AntsInFields: Stochastic simulation and bayesian inference for gibbs fields. *Springer electronic media*, 2003.

W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov Chain Monte Carlo in Practice.* Interdisciplinary Statistics. Chapman & Hall, London, Weinheim, New York, Tokyo, Melbourne, Madras, 1996.

J. Gutknecht. Oberon system 3: Vision of a future software technology. *Software-Concepts and Tools*, 15:26–33, 1994.

F. Hirzebruch and W. Scharlau. *Einführung in die Funktionalanalysis.* Spektrum Akademischer Verlag, Heidelberg, Berlin, Oxford, 1996.

E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Z. Physik*, 31:253, 1925.

A. Kempe. *Statistical analysis of Discontinuous Phenomena with Potts Functionals.* PhD thesis, Institute of Biomathematikcs and Biometry, National Research Center for Environment and Health, Munich, Germany, 2003.

J. Koplowitz, M. Lindenbaum, and A. Bruckstein. The number of digital straight lines on an $n \times n$ grid. *IEEE Trans. Inform. Theory*, 36(1):192–197, 1990.

S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, London, 1998.

V. Mottl, A. Kostin, and A. Kopylov. Edge preserving in generalized smoothing of signals and images. *Proceedings of the 14th International Conference on Pattern Recognition*, II:1579–1581, August 1998.

V. Mottl and I Muchnik. Serial and tree-serial dynamic programming with application to face identification. *DIMACS Technical Report 2002*, (45), 2002.

E. Le Pennec and S. Mallat. Sparse geometric image representation using bandelets. *IEEE IP*, to appear.

R.B. Potts. Some generalized order-disorder transitions. *Proc. Camb. Phil. Soc.*, 48:106–109, 1952.

K.T. Rockafellar and R.J-B. Wets. *Variational Analysis*. Springer Verlag, Berlin, Heidelberg, 2004.

J. Romberg, M. Wakin, and R. Baraniuk. Multiscale wedgelet image analysis: Fast decompositions and modeling. *IEEE International Conference on Image Processing*, September 2002.

A. Rosenfeld and R. Klette. Digital straightness. *Electronic Notes in Theoretical Computer Science*, 46, 2001.

W. Rudin. *Real and Complex analysis*. McGraw-Hill International Editions, Singapore, third edition, 1987.

G. Sawitzki. Extensible Statistical Software: On a Voyage to Oberon. *J. Computational and Graphical Statist.*, 5(3):263–283, 1996.

J. L. Starck, E. J. Candeès, and D. L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11:670–684, 2000.

Charles J. Stone. Optimal global rates of convergence for nonparametric regression. *Ann. Stat.*, 10:1040–1053, 1982.

C. Szyperski. *Component Software: beyond object-oriented programming*. ACM Press in co-operation with Addison-Wesley, 1998, 1998.

L.N. Trefethen and D. Bau. *Numerical linear algebra*. SIAM, Philadelphia, Pennsylvania, 1997.

K. Wagner and G. Wechsung. *Computational complexity.* Mathematics and its Applications (East European Series), 21. Dordrecht etc.: D. Reidel Publishing Company, a member of the Kluwer Academic Publishers Group (orig. publ. by VEB Deutscher Verlag der Wissenschaften, Berlin), 1986.

K. Wicker. *Das Potts Modell zur Segmentation verrauschter Daten*. Diploma thesis, Technische Universität München, 2004.

R.M. Willet. http://www.ece.rice.edu/∼willett/research/publications.html. *website*, 2004.

R.M. Willett, R. Castro, and R. Nowak. Coarse-to-fine manifold learning. *ICASSP*, 2004.

R.M. Willett and R.D. Nowak. Platelets: a multiscale approach for recovering edges and surfaces in photon-limited medical imaging. *IEEE MI*, 22:332–350, 2003.

G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*, volume 27 of *Applications of Mathematics*. Springer Verlag, Berlin, Heidelberg, New York, second edition edition, 2002.

G. Winkler and V. Liebscher. Smoothers for discontinuous signals. *J. Nonpar. Statist.*, 14(1-2):203–222, 2002.

N. Wirth and J. Gutknecht. *Project Oberon : the design of an operating system and compiler*. ACM Press, New York etc., 1992.