



UNIVERSITÄT ZU LÜBECK
INSTITUTE OF MATHEMATICS
AND IMAGE COMPUTING

The Rudin-Osher-Fatemi model and a hierarchical (BV,L2) image decomposition for multiscale image representation

Das Rudin-Osher-Fatemi Modell und eine hierarchische (BV,L2) Bildzerlegung im Hinblick auf multiskale Bildrepräsentation

Bachelorarbeit

verfasst am
Institute of Mathematics and Image Computing

im Rahmen des Studiengangs
Mathematik in Medizin und Lebenswissenschaften
der Universität zu Lübeck

vorgelegt von
Ole Gildemeister

ausgegeben und betreut von
Prof. Dr. Jan Modersitzki

Lübeck, den 30. November 2021

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ole Gildemeister

Zusammenfassung

Im Jahr 1992 stellten Rudin, Osher und Fatemi ein Verfahren zur Zerlegung von Bildern $f = u_\lambda + v_\lambda$ durch Minimierung ihrer Totalvariation vor [20]. Ursprünglich aus dem Feld der Rauschunterdrückung stammend, wird das so genannte *ROF Modell* seitdem vielfach in verschiedenen Anwendungen der Bildverarbeitung verwendet. Tadmor, Nezzar und Vese schlugen eine Erweiterung des Verfahrens zu einer hierarchischen Zerlegung vor, die zu einer multiskalen Bildrepräsentation $f \sim \sum_\ell u_\ell$ führt [22]. In dieser werden die verschiedenen Komponenten von f auf unterschiedliche Instanzen aufgeteilt: Beginnend mit einer stark vereinfachten, „Cartoon ähnlichen“ Version des Bildes, wird dessen Textur schrittweise erfasst, bis das Ausgangsbild schließlich wiederhergestellt ist. In dieser Arbeit gebe ich einen Überblick über das *ROF Modell* sowie die hierarchische (BV, L^2) Zerlegung und diskutiere ihre Eignung hinsichtlich der Repräsentation von 2D Graustufenbildern. Zudem stelle ich ein neues Verfahren zur Initialisierung der hierarchischen Zerlegung vor, durch das Elemente von möglicherweise fehlenden Skalen erfasst werden.

Abstract

In 1992, Rudin, Osher and Fatemi introduced a method for decomposing images $f = u_\lambda + v_\lambda$ by minimising their total variation [20]. Originating from the field of denoising, this so-called *ROF model* is now prominently used in various applications in image processing. Tadmor, Nezzar and Vese proposed to extend this method to a hierarchical decomposition, yielding a multiscale image representation $f \sim \sum_\ell u_\ell$ [22]. In this, the different components of f are captured at different instances: Starting with a very simplified, “cartoonish” version of the image, its texture is resolved step by step, until the original image is finally reconstructed. In this thesis, I give an overview of the *ROF model* as well as the hierarchical (BV, L^2) decomposition and discuss its applicability for the representation of 2D greyscale images. Furthermore, I introduce a new initialisation method to the hierarchical decomposition by which elements of potentially missing scales are captured.

Contents

1	Introduction and motivation	1
2	Total variation and the BV spaces	3
2.1	Definition of total variation and the L^2 and BV function spaces	3
2.2	Total variation of differentiable and weakly differentiable functions	4
2.3	Total variation of functions in one variable	6
2.4	Properties of total variation and functions of bounded variation	8
3	The Rudin-Osher-Fatemi model	10
3.1	Definition and description of the ROF model	10
3.2	A selection of properties of the ROF model	11
3.3	The Euler-Lagrange equation of the ROF functional	17
4	Extension of the ROF model to a hierarchical (BV, L^2) decomposition	18
4.1	Description of the hierarchical decomposition	18
4.2	Convergence of the hierarchical decomposition	19
4.3	Initialisation of the hierarchical decomposition	23
5	Discretisation and implementation of the models	25
5.1	Notation and Grids	25
5.2	Implementation of the ROF model	26
5.3	Implementation of the hierarchical decomposition	34
6	Numerical results	36
6.1	Results on the one-time ROF decomposition	36
6.2	Results on the hierarchical (BV, L^2) image decomposition	39
7	Conclusion	45
	Bibliography	46

1

Introduction and motivation

Starting point of this thesis is the so-called *ROF model*, named after Rudin, Osher and Fatemi who in 1992 proposed a method for denoising images by minimising their total variation [20]. The basic idea is to decompose a 2D greyscale image $f \in L^2(\mathbb{R}^2)$ into two components $f = u + v$ where $u \in BV(\mathbb{R}^2)$ represents a “cartoonish” version of the image, and the residual $v \in L^2(\mathbb{R}^2)$ contains its oscillatory part like noise and textures.

The model gained use in many applications of image processing, see e.g. [4], also because the resultant u is an element of the in many contexts more “well-behaving” function space of bounded variation $BV(\mathbb{R}^2) \subset L^2(\mathbb{R}^2)$ [cf. 16, page 14] and has some desirable properties: BV functions are “measure theoretically piecewise continuous” only with “jumps along a measure theoretically C^1 surface” [11, Section 5.9] and may - at least in some sense - be considered differentiable [cf. 11, Section 6.1.3]. On the other hand and unlike functions from the *Sobolev space* $W^{1,2}(\mathbb{R}^2)$, they are also capable of preserving edges and thus representing information of an image more properly [4, pages 266f]. The concept behind the *ROF model* was also that groundbreaking because the minimisation of total variation naturally reduces the noise or more generally the occurrence of “spurious oscillations” [20] in an image, and yields a relatively simple and smoothened result.

In the *ROF model*, the question how many details of the original image f are preserved in u , is determined by the choice of a weighting parameter $\lambda > 0$: While small λ result in a relatively “coarse” u , greater λ penalise the *fidelity term* $\|v\|_2^2$ stronger and hence u contains more information of f . In view of this, the parameter λ can also be interpreted as a *cut-off scale* [22]: It serves as a scaling level determining which elements of f are interpreted as texture and how much f should be simplified. This lead to the idea of iteratively decomposing an image with an exponentially increasing parameter $\lambda_\ell = 2^\ell \lambda_0$, or more precisely decomposing the remaining residual $v_{\ell-1}$ for $\ell = 1, \dots, m$.

Using this *hierarchical* (BV, L^2) *decomposition*, in the end one obtains a multiscale representation $f = \sum_{\ell=0}^m u_\ell + v_m$, as first introduced by Tadmor, Nezzar and Vese in 2004 [22]. This concept got recent attention when earlier this year in [9], it was transferred to the application of multiscale registration, going from a “coarse” level of global structural deformations to more locally refined ones.

This thesis gives an introduction to the *ROF model* and studies the work of [22] on hierarchical (BV, L^2) decompositions of 2D greyscale images. To this end, I will start in Chapter 2 by defining total variation and the BV space and name some of their properties. Next in Chapter 3, I describe the *ROF model* in detail and present some of its characteristics. There, the focus lies on the interpretation of the model as separation of texture from the “essential / cartoonish” features of an image, and on the impact of the tuning parameter λ to the result. In this context, Theorem 3.4 from Meyer [16] plays a major role, by which λ determines whether elements end up in u_λ and v_λ .

Thereafter, I extend this method to the aforesaid *hierarchical* (BV, L^2) *decomposition* in Chapter 4 and discuss, why the resulting multiscale description of an image is suitable for its representation. Furthermore, I give a new initialisation method to capture possibly missing larger scales in Section 4.3, inspired by the original proposal of [22].

In the last part, I describe a way to numerically tackle both the *ROF model* and its hierarchical application for obtaining a multiscale representation of real images. The theoretical considerations and

algorithms are discussed in Chapter 5. Results on their performance and actual decompositions of images are presented in Chapter 6. The code I implemented for the thesis, is written and executed in MATLAB (Version R2020b) [15] and published on GitHub [13]. As main example, I will use a photograph of the *Aurlandsfjord* in Norway taken by myself in June 2021. It is shown in Figure 1.1 and from here on will be referred to as *fjord picture*.

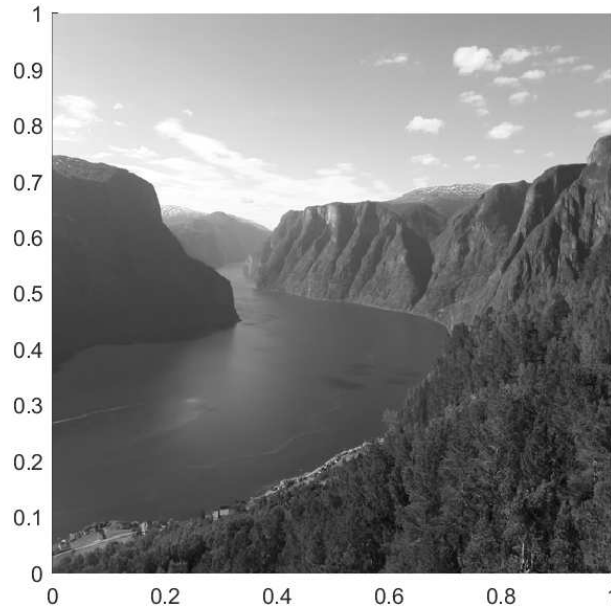


Figure 1.1: “Fjord picture” – This greyscaled photograph of the *Aurlandsfjord* in Norway serves as an example and test image for most of the thesis.

2

Total variation and the BV spaces

2.1 Definition of total variation and the L^2 and BV function spaces

At first, I will give some preliminaries which will be needed for the further discussions in this thesis. In general, I follow on [22] and consider images as L^2 objects. More precisely, the thesis focusses on 2D greyscale images which are here given by $f \in L^2(\mathbb{R}^2)$. Hence, I start by introducing the L^p space over real domains together with its associated norm.

Definition 2.1. (following [10, Chapter 6 Definition 2.1] and [11]) Let $n \in \mathbb{N}$ and $\Omega \subseteq \mathbb{R}^n$ be an open set, $p \geq 1$ and λ denote the *Lebesgue measure* [10]. For a λ -measurable function $f : \Omega \rightarrow \mathbb{R}$, the L^p -**norm** is defined as

$$\|f\|_p := \|f\|_{L^p(\Omega)} := \left(\int_{\Omega} |f|^p d\lambda \right)^{\frac{1}{p}}. \quad (2.1)$$

The function space that consists of all functions f with $\|f\|_p < \infty$, is called L^p -**space** over Ω and denoted by $L^p(\Omega)$. In the following, mainly L^1 and L^2 will be of interest.

Furthermore, for two functions $f, g \in L^2(\Omega)$, a *scalar product* [cf. 10, page 255] is defined by

$$\langle f, g \rangle := \int_{\Omega} fg d\lambda. \quad (2.2)$$

Remark. In fact, in the context of the above defined spaces, these “norms” are only *semi-norms* because not only the zero function $\phi = 0$ has “norm” 0. To obtain proper norms, all functions that coincide *almost everywhere* need to be identified with each other, so the norms are actually defined on equivalent classes of functions [cf. 10, page 250f]. However, that distinguishment is not relevant here.

The main concept behind the *ROF model* and hence this thesis’ topic is to decompose an image $f \in L^2(\Omega)$ by minimising its total variation. Therefore, at next I define this quantity. Note that the total variation is usually defined for L^1 -integrable functions and there is no general inclusion relation between L^1 and L^2 [10, page 253], so to be precise, in the context of this thesis images must be assumed as $f \in L^2(\Omega) \cap L^1(\Omega)$. The next two definitions follow [11, Definition 5.1] and [4, Definition 1.1]:

Definition 2.2. Let $n \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set and $f \in L^1(\Omega)$. The **total variation of f in Ω** is defined as

$$\text{TV}(f, \Omega) := \sup_{\substack{\varphi \in C_c^1(\Omega, \mathbb{R}^n) \\ \|\varphi\|_{L^\infty(\Omega)} \leq 1}} \int_{\Omega} f(x) \operatorname{div}(\varphi(x)) dx \quad (2.3)$$

where $C_c^1(\Omega, \mathbb{R}^n)$ denotes the space of continuously differentiable functions from Ω to \mathbb{R}^n with compact support, $\|\varphi\|_{L^\infty(\Omega)}$ the essential supremum norm (see e.g. [10, page 243]) and $\operatorname{div}(\varphi) := \sum_{i=1}^n \frac{\partial \varphi_i}{\partial x_i}$ the divergence of φ .

Definition 2.3. Let $n \in \mathbb{N}$ and $\Omega \subseteq \mathbb{R}^n$ be an open set.

A function $f \in L^1(\Omega)$ is said to be of **bounded variation** (BV function) if its total variation $\text{TV}(f, \Omega)$ is finite. Thus, the **space of functions of bounded variation (BV space)** is defined as

$$BV(\Omega) := \{f \in L^1(\Omega) \mid \text{TV}(f, \Omega) < \infty\}. \quad (2.4)$$

For $f \in BV(\Omega)$, the total variation yields a semi-norm which from now on will be denoted by

$$\|f\|_{BV} := \|f\|_{BV(\Omega)} := \text{TV}(f, \Omega) \quad (2.5)$$

Remark. a) $\|\cdot\|_{BV}$ fails to be a norm because it is 0 not only for $\phi = 0$: As will be discussed in Lemma 2.7, for any constant function $\psi = c$ it holds $\|\psi\|_{BV} = 0$. However, proper norms on BV can be found that give rise to the latter being a *Banach space* [4, page 273].

b) By definition it holds $BV(\Omega) \subseteq L^1(\Omega)$, and quoting [16, page 24] in fact $BV(\Omega) \subset L^{\frac{n}{n-1}}(\Omega)$. So in particular, for the context of $n = 2$ that means $BV(\Omega) \subset L^2(\Omega)$.

2.2 Total variation of differentiable and weakly differentiable functions

For continuously differentiable functions, the total variation simplifies to another form which is also basic for its discretisation in Chapter 5:

Theorem 2.4. [1] *Let $n \in \mathbb{N}$ and $\Omega \subseteq \mathbb{R}^n$ be an open set. If a function $f : \Omega \rightarrow \mathbb{R}$ is continuously differentiable, i.e. $f \in C^1(\Omega)$, then the total variation is given by*

$$\text{TV}(f, \Omega) = \int_{\Omega} |\nabla f(x)| \, dx = \|\nabla f\|_1 \quad (2.6)$$

where $|\cdot|$ denotes the Euclidean vector norm.

This statement is mentioned and used a lot in the literature, see e.g. [4, page 271], [6, page 2] or [12, page 75]. And in fact, it is not limited to continuously differentiable functions, but also works for *weakly differentiable* functions. Then, the gradient ∇f is replaced by the *weak gradient* Df – a generalisation of the gradient’s concept which is defined next following [11, Section 4.1]:

Definition 2.5. Let $I \subseteq \mathbb{R}$ be an open interval. A univariate function $f : I \rightarrow \mathbb{R}$ is called **weakly differentiable** if there exists a function $g \in L^1(I)$ with

$$\int_I f(x)\varphi'(x)dx = - \int_I g(x)\varphi(x) \, dx \quad \forall \varphi \in C_c^1(I).$$

Then, g is called a **weak derivative of f** .

Now let $n \in \mathbb{N}$ and $\Omega \subseteq \mathbb{R}^n$ be an open set. A multivariate function $f \in L^1(\Omega)$ is called **weakly differentiable** if it is weakly differentiable with respect to every component, that is if there exist functions $g_1, \dots, g_n \in L^1(\Omega)$ with

$$\int_{\Omega} f(x)\varphi_{x_i}(x)dx = - \int_{\Omega} g_i(x)\varphi(x) \, dx \quad \forall \varphi \in C_c^1(\Omega), \quad i = 1, \dots, n. \quad (2.7)$$

Then, g_i is called **weak partial derivative of f w.r.t x_i** and the vector $Df := (g_1, \dots, g_n)$ **weak gradient of f** .

Remark. The definitions of weak partial derivative and weak gradient are consistent in the following sense: For $f \in C^1(\Omega)$ they coincide with the “actual” partial derivative f_{x_i} and the gradient ∇f . This can be seen by integrating (2.7) by parts and using the compact support of φ by which it must be 0 at the boundary $\partial\Omega$ of an open set Ω :

$$\int_{\Omega} f(x)\varphi_{x_i}(x)dx = f(x)\varphi(x)\Big|_{\partial\Omega} - \int_{\Omega} f_{x_i}(x)\varphi(x) dx \stackrel{\varphi \in C_c^1(\Omega)}{=} - \int_{\Omega} f_{x_i}(x)\varphi(x) dx.$$

Thus, the partial derivative f_{x_i} is a weak partial derivative of f w.r.t x_i (and by [11, page 144], that weak partial derivative is also unique almost everywhere).

Now, a generalisation of Theorem 2.4 for functions f with L^1 -integrable weak gradient, that is $\| |Df| \|_1 < \infty$, is given as follows:

Lemma 2.6. [cf. 11, pages 197f] *Let $n \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set and $f : \Omega \rightarrow \mathbb{R}$ be weakly differentiable. If $\| |Df| \|_1 < \infty$, then $f \in BV(\Omega)$ and its total variation is given by*

$$\text{TV}(f, \Omega) = \int_{\Omega} |Df(x)| dx = \| |Df| \|_1 \tag{2.8}$$

where $|\cdot|$ denotes the Euclidean vector norm.

Clearly, Theorem 2.4 and Lemma 2.6 only describe special cases for computing the total variation of a function f . However, to stress the relation between weak gradient and total variation (or their corresponding measures, see [4, Section 1.5]), the latter is sometimes (e.g. in [4], [14], [12]) also in general denoted by

$$\|f\|_{BV} = \text{TV}(f, \Omega) = \int_{\Omega} |Df|.$$

Furthermore, the concept for a discrete approximation to the total variation carried out in Section 5.1 arises from eq. (2.8), as well as the form of the *Euler-Lagrange equation* to the later defined *ROF problem*. That equation is introduced in Section 3.3 and basis for the numerical methods of Section 5.2.

Next, I will give an intuitively desirable result: Constant functions have a total variation of zero. More precisely, this holds for any function $f : \Omega \rightarrow \mathbb{R}$ which is constant *almost everywhere*, meaning that it takes the same real value on the entire domain except for at most on a *Lebesgue null set* $N \subset \Omega$ – a set with Lebesgue measure $\lambda(N) = 0$ [cf. 10, Chapter 4 Definition 4.1].

When dealing with images on bounded domains and non-zero mean, this idea will be important as it allows for manipulating the image by adding a constant signal without changing its total variation. More on that in Section 3.2.

Lemma 2.7. *Let $n \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set, $c \in \mathbb{R}$ and $f : \Omega \rightarrow \mathbb{R}$ with $f(x) = c$ almost everywhere. The (possibly empty) null set on which $f(x) \neq c$ be denoted by $N \subset \Omega$. Then, the total variation of f is given by $\text{TV}(f, \Omega) = 0$.*

Proof. This follows directly from Definition 2.2 and the divergence theorem because for any $\varphi \in C_c^1(\Omega, \mathbb{R}^n)$ it holds by the compact support of φ (which imposes $\varphi = 0$ on $\partial\Omega$):

$$\int_{\Omega} f \operatorname{div}(\varphi) d\lambda = \int_{\Omega \setminus N} f \operatorname{div}(\varphi) d\lambda = c \int_{\Omega \setminus N} \operatorname{div}(\varphi) d\lambda = c \int_{\Omega} \operatorname{div}(\varphi) d\lambda = c \int_{\partial\Omega} \varphi \cdot n d\lambda = 0.$$

Now when taking the supremum over φ , one obtains $\text{TV}(f, \Omega) = 0$.

Alternatively, by the same argument the weak gradient of f must be given by $Df = (0, \dots, 0)$, and eq. (2.8) yields $\text{TV}(f, \Omega) = 0$. \square

2.3 Total variation of functions in one variable

To get a better understanding of the total variation, the one-dimensional case will be considered now. Unfortunately, the term *total variation* is not distinct, and especially in 1D, there exist various, non-necessarily equivalent definitions of the total variation. In this thesis, TV is given as in Definition 2.2.

However, there is a second, commonly used definition for total variation in which the domain is divided into a partition and the difference between all neighbouring function values is maximised. This quantity will be used to compute the total variation in Example 2.10 a) – which works because it coincides with TV when $f \in C^1$, see e.g. [3, Definition 1.1]:

Lemma 2.8. *Let $[a, b] \in \mathbb{R}$ be a real interval and $P = \{t_0, \dots, t_m\}$ a partition of $[a, b]$, i.e. $a =: t_0 < \dots < t_m := b$. If the function $f : [a, b] \rightarrow \mathbb{R}$ is continuously differentiable on (a, b) , then the total variation of f can also be described by the following term:*

$$\text{TV}(f, \Omega) = \int_a^b |f'(x)| dx = \sup_{P \in \mathcal{P}([a, b])} \sum_{j=1}^m |f(t_j) - f(t_{j-1})|. \quad (2.9)$$

where $\mathcal{P}([a, b])$ denotes the family of partitions of $[a, b]$.

Next, I want to give three examples of 1D-functions to illustrate some aspects in the relation between total variation, continuity and the (squared) L^2 -norm. In my opinion, they give a good impression on the total variation and why its minimisation can be desirable when seeking a representation for signals (such as images). Example 2.9 shows a sequence of signals $(f_n)_{n \in \mathbb{N}}$ which all have the same energy (in an L^2 -sense), but the total variation diverges corresponding to the increasing number of oscillations in f_n :

Example 2.9. Be $n \in \mathbb{N}$ and $f_n : [0, 2\pi] \rightarrow \mathbb{R}$ with $f_n(x) := \sin(nx)$. It holds:

$$\begin{aligned} \|f_n\|_2^2 &= \int_0^{2\pi} (f_n(x))^2 dx = \int_0^{2\pi} \sin^2(nx) dx \\ &= \frac{1}{2} \int_0^{2\pi} (1 - \cos(2nx)) dx \\ &= \frac{1}{2} \left(x - \frac{1}{2n} \sin(2nx) \right) \Big|_0^{2\pi} = \pi, \end{aligned}$$

$$\begin{aligned} \|f_n\|_{BV} &= \int_0^{2\pi} |f_n'(x)| dx = \int_0^{2\pi} |n \cos(nx)| dx \\ &= \underbrace{\int_0^{\frac{\pi}{2n}} n \cos(nx) dx}_{=1} + \sum_{i=0}^{n-1} \underbrace{\int_{(i+\frac{1}{4})\frac{2\pi}{n}}^{(i+\frac{3}{4})\frac{2\pi}{n}} -n \cos(nx) dx}_{=2} \\ &\quad + \sum_{i=1}^{n-1} \underbrace{\int_{(i-\frac{1}{4})\frac{2\pi}{n}}^{(i+\frac{1}{4})\frac{2\pi}{n}} n \cos(nx) dx}_{=2} + \underbrace{\int_{2\pi-\frac{\pi}{2n}}^{2\pi} n \cos(nx) dx}_{=1} \\ &= 1 + 2n + 2(n-1) + 1 = 4n. \end{aligned}$$

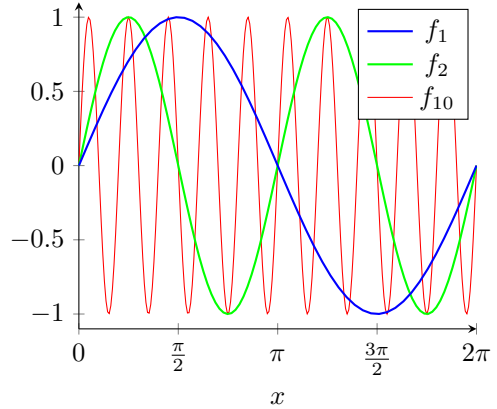


Figure 2.1: Visualisation of $f_n(x) = \sin(nx)$ for $n = 1, 2, 10$. All signals have the same energy, but the increasing number of oscillations for larger n leads to an increase in total variation.

Thus, $f_n \in BV([0, 2\pi])$. As one can see, the squared L^2 -norm remains the same for all n , whereas the variation increases linearly with the number of oscillations taking place. This idea of higher oscillating signals leading to a larger total variation in my opinion is basic for the concept of image denoising by total variation minimisation and in the end also the multiscale image representation discussed in this thesis: That is because noise as well as texture elements of an image are usually associated with highly oscillating signal information [22]. More on that in Chapter 3 and Chapter 4.

The other two functions – presented in Example 2.10 – are modifications of the signal $\sin(1/x)$. Both have finite energy, are continuous in $[0, 1]$ and even continuously differentiable in $(0, 1]$, but whereas the total variation of the first function is infinite, the second one is sufficiently damped to be of bounded variation. In my interpretation, these signals show again how the quantity of total variation depends on the frequency of oscillations, but also on the size of the amplitude associated with the oscillations.

To compute the total variation in Example 2.10 a), I use Lemma 2.8 which works since the function is continuously differentiable in $(0, 1)$.

Example 2.10.

a) Be $f : [0, 1] \rightarrow \mathbb{R}$, $f(x) := \begin{cases} 0 & \text{for } x = 0 \\ x \sin\left(\frac{1}{x}\right) & \text{for } x \in (0, 1] \end{cases}$.

Since $\lim_{x \rightarrow 0} |x \sin\left(\frac{1}{x}\right)| \leq \lim_{x \rightarrow 0} |x| = 0$, f is continuous in $[0, 1]$. Furthermore, f is continuously differentiable in $(0, 1]$ with $f'(x) = \sin\left(\frac{1}{x}\right) - \frac{1}{x} \cos\left(\frac{1}{x}\right)$.

For the energy of f , it holds:

$$\begin{aligned} \|f\|_2^2 &= \int_0^1 (f(x))^2 dx = \int_0^1 x^2 \sin^2\left(\frac{1}{x}\right) dx \\ &\leq \int_0^1 x^2 dx = \frac{1}{3} x^3 \Big|_0^1 = \frac{1}{3}. \end{aligned}$$

Following [3, Example 1.8], to compute the total variation of f on $[0, 1]$, the partition $P = \{t_0, \dots, t_m\}$ is chosen such that t_{j-1} and t_j in (2.9) are alternating maxima and minima of f . This is achieved by:

$$t_0 := 0, \quad t_1 := \frac{2}{(2m-1)\pi}, \quad \dots, \quad t_j := \frac{2}{(2m-2j+1)\pi}, \quad \dots, \quad t_{m-1} := \frac{2}{3\pi}, \quad t_m := 1.$$

Now it holds $|f(t_j) - f(t_{j-1})| = \left| \sin\left(\left(m-j+\frac{1}{2}\right)\pi\right) t_j - \sin\left(\left(m-j+\frac{3}{2}\right)\pi\right) t_{j-1} \right| \geq 2t_{j-1}$ for $j = 2, \dots, m-1$, therefore

$$\text{TV}(f, (0, 1)) \geq \sum_{j=1}^m |f(t_j) - f(t_{j-1})| \geq \sum_{j=2}^{m-1} 2t_{j-1} = \frac{2}{\pi} \sum_{j=2}^{m-1} \frac{2}{2m-2j+3} = \frac{2}{\pi} \sum_{j=3}^m \frac{2}{2j-1} \geq \frac{2}{\pi} \sum_{j=3}^m \frac{1}{j}.$$

Letting $m \rightarrow \infty$ now yields $\text{TV}(f, (0, 1)) \geq \lim_{m \rightarrow \infty} \frac{2}{\pi} \sum_{j=3}^m \frac{1}{j} = \infty$ by the divergence of the harmonic series.

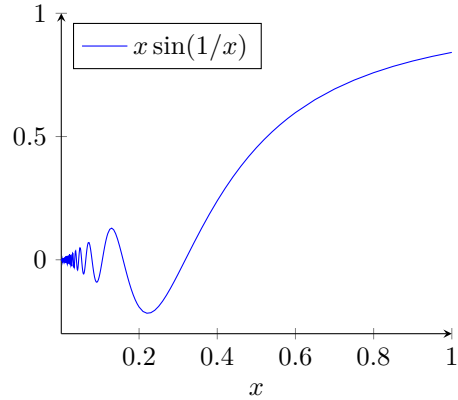


Figure 2.2: Visualisation of the signal $f(x) = x \sin(1/x)$. It is continuously differentiable and has finite energy, but infinite total variation.

$$\text{b) Be } f : [0, 1] \rightarrow \mathbb{R}, f(x) := \begin{cases} 0 & \text{for } x = 0 \\ x^2 \sin\left(\frac{1}{x}\right) & \text{for } x \in (0, 1] \end{cases}.$$

Again, f is continuous in $[0, 1]$, and continuously differentiable in $(0, 1]$ with $f'(x) = 2x \sin\left(\frac{1}{x}\right) - \cos\left(\frac{1}{x}\right)$.

An upper bound for the energy of f is now given by:

$$\begin{aligned} \|f\|_2^2 &= \int_0^1 (f(x))^2 dx = \int_0^1 x^4 \sin^2\left(\frac{1}{x}\right) dx \\ &\leq \int_0^1 x^4 dx = \frac{1}{5} x^5 \Big|_0^1 = \frac{1}{5}. \end{aligned}$$

And in difference to a), here the total variation can be estimated by:

$$\begin{aligned} \|f\|_{BV} &= \int_0^1 |f'(x)| dx = \int_0^1 \left| 2x \sin\left(\frac{1}{x}\right) - \cos\left(\frac{1}{x}\right) \right| dx \\ &\leq \int_0^1 \left(\underbrace{\left| 2x \sin\left(\frac{1}{x}\right) \right|}_{\leq 2x} + \underbrace{\left| \cos\left(\frac{1}{x}\right) \right|}_{\leq 1} \right) dx \leq x^2 + x \Big|_0^1 = 2. \end{aligned}$$

Hence, in both cases $f \in L^2([0, 1])$, $f \in C^0([0, 1])$ and even $f \in C^1((0, 1])$, so the multiplication by x resp. x^2 especially yields continuity. But whereas x does not suffice to limit the total variation of f in case a), the “stronger damping” induced by the x^2 -term in case b), does not only ensure continuity, but also bounds the variation effectively.

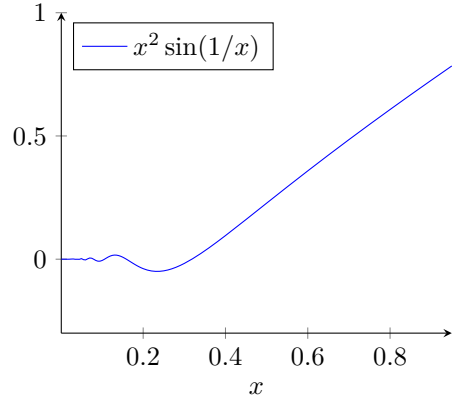


Figure 2.3: Visualisation of the signal $f(x) = x^2 \sin(1/x)$. It is continuously differentiable, has finite energy and is of bounded variation.

2.4 Properties of total variation and functions of bounded variation

Now back to the multivariate case: Besides Theorem 2.4 resp. Lemma 2.6, there is a great number of properties regarding the quantity of total variation as well as the BV space in general, see e.g. [11], [3] or [4]. They give an idea to the question, why signals of bounded variation might suit for representing information of images, but a detailed discussion on that goes beyond the scope of this thesis.

Instead, I want to state only two properties of the total variation, namely its *lower semi-continuity* and its *convexity*. Both will be needed later in Section 3.2 when discussing the existence and uniqueness of a minimiser to the (ROF) problem. Afterwards, I will quote two more – in my opinion very important – results on BV functions.

Lemma 2.11. [cf. 4, Section 1.2.3]

a) *The total variation is **lower semi-continuous**, that is:*

Let $\Omega \subseteq \mathbb{R}^n$ for $n \in \mathbb{N}$ be an open set and $f, f_k \in L^1(\Omega)$ for $k \in \mathbb{N}$ where $f_k \rightarrow f$ in $L^1(\Omega)$. Then it holds:

$$\text{TV}(f, \Omega) \leq \liminf_{k \rightarrow \infty} \text{TV}(f_k, \Omega).$$

b) *The total variation is **convex**, i.e. for all $f_1, f_2 \in L^1(\Omega)$ and $t \in [0, 1]$ it holds:*

$$\text{TV}(tf_1 + (1-t)f_2, \Omega) \leq t \text{TV}(f_1, \Omega) + (1-t) \text{TV}(f_2, \Omega).$$

Proof. The second statement follows directly from the supremum in (2.3) and the linearity of integrals. For the first statement, the continuity of the integral form imposes for any $\varphi \in C_c^1(\Omega, \mathbb{R}^n)$ with $\|\varphi\|_{L^\infty(\Omega)} \leq 1$:

$$\int_{\Omega} f \operatorname{div}(\varphi) = \lim_{k \rightarrow \infty} \int_{\Omega} f_k \operatorname{div}(\varphi) \leq \liminf_{k \rightarrow \infty} \text{TV}(f_k, \Omega).$$

When taking the supremum over φ , the left hand side becomes the definition of total variation in (2.3) and the lower semi-continuity follows. \square

Finally, I come to the two properties of the BV space that I want to point out. In my opinion, both give important information on the central question in this thesis, why minimising the total variation might be desirable when seeking a suitable representation for an image. Furthermore, they clarify the impact of a potentially small total variation in a signal. For more background and the proofs, see the indicated references.

The first statement gives a result on the approximation of BV functions: It ensures that any function $f \in BV(\Omega)$ and its total variation can be locally approximated by a sequence of smooth functions. This is important especially when numerically dealing with the *ROF problem*, since the methods from Section 5.2 to discretely approximate the minimiser mainly treat the image as if it was continuously differentiable, as will be discussed in that section.

The second result yields information on the total variation of BV functions: On the one hand, it generally serves as an upper bound of their size in an L^1 (and for $n = 2$ even L^2)-sense. And on the other hand, it also limits the “difference” between the function and its mean in every ball (and hence in particular in any domain). In my opinion, especially that second aspect also clarifies the meaning of the term *total variation*.

Lemma 2.12.

- a) [cf. 11, Theorem 5.3] *Let $n \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set and $f \in BV(\Omega)$ be a function of bounded variation. There exists a set of infinitely differentiable functions with bounded variation $\{f_k\}_{k=1}^\infty \subseteq BV(\Omega) \cap C^\infty(\Omega)$ such that*

$$f_k \rightarrow f \text{ in } L^1(\Omega) \text{ and } \|f_k\|_{BV} \rightarrow \|f\|_{BV} \text{ as } k \rightarrow \infty.$$

- b) [cf. 11, Theorem 5.10] *Be $n \in \mathbb{N}$. There exists a constant $C_1(n)$ such that for all functions of bounded variation $f \in BV(\mathbb{R}^n)$:*

$$\|f\|_{L^{\frac{n}{n-1}}(\mathbb{R}^n)} \leq C_1(n) \|f\|_{BV(\mathbb{R}^n)}.$$

Further, there exists a second constant $C_2(n)$ such that for all balls $B_r(y) \subset \mathbb{R}^n$ with radius $r > 0$ and centre $y \in \mathbb{R}^n$ and for all $f \in BV(\mathbb{R}^n)$ it holds:

$$\|f - \bar{f}\|_{L^{\frac{n}{n-1}}(B_r(y))} \leq C_2(n) \|f\|_{BV(B_r(y))},$$

where $\bar{f} := \int_{B_r(y)} f(x) dx$ denotes the “mean” of f on $B_r(y)$.

3

The Rudin-Osher-Fatemi model

The decomposition of an image by minimising its total variation originates from the field of image denoising or, more generally, image reconstruction [20]. As the name indicates, this process aims at reconstructing the original signal u or something as close to that as possible from a given defective signal f . Hence, one seeks a decomposition $f = u + v$ where u represents the desired “original” signal and $v = f - u$ the residual. In a perfect scenario, v would obviously be exactly the error (e.g. noise).

In this setting, an approximation to u is defined as a minimiser of a properly chosen energy functional $E : L^2(\Omega) \rightarrow \mathbb{R}$ of the form

$$E(u) = F(u) + \lambda \|f - u\|_2^2.$$

where $F : L^2(\Omega) \rightarrow \mathbb{R}$ is a functional corresponding to the type of signal to reconstruct and specifies by what characteristics u should be evaluated [cf. 4, Section 1.1.2]. Of particular interest is the tuning parameter $\lambda > 0$ which determines the weight of both components in E , and will later be discussed in detail for the *ROF model*.

Due to the authors of [4], “a good F should simultaneously ensure some spatial regularity, but also preserve the edges” [4, page 267]. In 1992, Rudin, Osher and Fatemi proposed in a seminal paper [20] to consider the *total variation* as regulariser, i.e. $F(u) = \text{TV}(u, \Omega)$. Since then, this idea is being referred to as *ROF model*. By construction, it results in a signal of bounded variation, i.e. $u \in BV(\Omega)$. In this chapter, I will give a precise and mathematical description of the model and discuss some of its properties.

3.1 Definition and description of the ROF model

In the context of this thesis, the *ROF model* describes the decomposition $f = u_\lambda + v_\lambda$ of a 2D greyscale image $f \in L^2(\Omega)$ into a “cartoonish” version $u_\lambda \in BV(\Omega)$ and the residual $v_\lambda \in L^2(\Omega)$. The decomposition is obtained by minimising the energy functional

$$\mathcal{E}_{f,\lambda} : BV(\Omega) \rightarrow \mathbb{R}, \quad \mathcal{E}_{f,\lambda}(u) := \|u\|_{BV} + \lambda \|f - u\|_2^2 \quad (3.1)$$

over all functions of bounded variation $u \in BV(\Omega)$ for a pre-specified weighting parameter $\lambda > 0$. Thus, the *ROF problem* is defined as the minimisation problem

$$\underset{u \in BV(\Omega)}{\text{minimise}} \quad \|u\|_{BV} + \lambda \|f - u\|_2^2. \quad (\text{ROF})$$

Here, $\lambda \|f - u\|_2^2$ is the fidelity term, $\|u\|_{BV}$ serves as regularisation and the parameter $\lambda > 0$ determines the influence of both terms to the minimisation process.

Notation: In this thesis, I will refer to the actual obtained infimum in (ROF) as

$$J(f, \lambda) := \inf_{u \in BV(\Omega)} \mathcal{E}_{f,\lambda}(u)$$

which measures how well f can be approximated by its “ BV -features” [cf. 22, page 555], and to the resultant arguments u_λ and v_λ as

$$[u_\lambda, v_\lambda] := \arg \min_{u+v=f} J(f, \lambda) := \left[\arg \min_{u \in BV(\Omega)} \mathcal{E}_{f,\lambda}(u), f - \arg \min_{u \in BV(\Omega)} \mathcal{E}_{f,\lambda}(u) \right]. \quad (3.2)$$

This last notation with the arg min of the J -functional $J(f, \lambda)$ will prove useful for clarity – especially in the discussion on the hierarchical decomposition in Chapter 4 – and follows the style of [22] and [9].

I want to point out that in their original paper, Rudin, Osher and Fatemi introduced a different version of the minimisation problem: As mentioned above, the *ROF model* has its origins in the image denoising. This problem underlies the assumption that a given signal $f \in L^2(\Omega)$ consists of the actual image $\hat{u} \in L^2(\Omega)$ and some additive noise $\hat{v} \in L^2(\Omega)$ of mean 0 and standard deviation σ , i.e. $f = \hat{u} + \hat{v}$ [20]. Thus, their procedure aimed at computing a decomposition $f = u + v$ such that u approximates \hat{u} as accurate as possible by minimising its total variation; the noise \hat{v} is analogously approximated by the residual $v = f - u$. In view of that, Rudin, Osher and Fatemi originally proposed to solve the constrained minimisation problem

$$\underset{u \in BV(\Omega)}{\text{minimise}} \quad \|u\|_{BV} \quad \text{subject to} \quad \int_{\Omega} (f - u) = 0 \quad \text{and} \quad \frac{1}{2} \|f - u\|_2^2 = \sigma^2. \quad (3.3)$$

Existence, uniqueness and some statements regarding the *well-posedness* of problem (3.3) were proven in 1994 by Acar and Vogel [2], and more generally three years later by Chambolle and Lions [5]. In the latter, it was also proven that solving (3.3) is equivalent to solving the unconstrained problem (ROF) for some parameter $\lambda > 0$, and this is today most commonly known as *ROF model*.

By the discussions of Chapter 2, one would expect the result $u_\lambda \in BV(\Omega)$ to be a simpler and smoother signal than f that consists of its essential features (especially large objects and edges), whereas v represents the oscillatory part such as spurious components or texture. This interpretation of the decomposition as distinguishment between *edges* and *texture* [16] [22] is fundamental for the later discussed multiscale representation.

In view of this, the parameter λ does not only weight the influence of both terms in $\mathcal{E}_{f,\lambda}$: It is a *cut-off scale* [22] that defines whether elements of f are rather interpreted by the *ROF model* as texture or not, and thus determines how much f should be simplified. As will be discussed in the next section, this question is closely related to the oscillatory character of an element.

It usually requires a priori information on the image to properly select λ : If it is chosen too small, only few information are kept in a “very simple” u_λ , whereas a great λ may produce a resultant u_λ that is very close to and contains “too many” details of the original image f [cf. 22, Section 2.1]. This is illustrated in Figure 3.1. In this thesis’ consideration of a hierarchical decomposition, however, that problem is only minor, as will be discussed in Section 4.3.

The *ROF model* can be (and often were and is) modified – for instance by using the non-squared L^2 -norm $\|f - u\|_2$ [cf. 16, page 29] or L^1 -norm $\|f - u\|_1$ [cf. 4, page 56] as fidelity term, and extended to more general applications such as *pre-transformed models* $f = Au + v$ where A is a linear operator and represents some transformation (e.g. blurring, sampling) [4]. In this thesis, however, I will only discuss the above defined (ROF) problem.

3.2 A selection of properties of the ROF model

As mentioned above, a prove for the existence of a minimiser of (ROF) was given in [5]. Nonetheless, I consider this to be such a central result for the discussion of this thesis that I want to sketch a proof following [4, Section 1.2.3]:

Considering a minimising sequence $(u_k)_{k \in \mathbb{N}}$ for $\mathcal{E}_{f,\lambda}$ such that $\mathcal{E}_{f,\lambda}(u_k) \rightarrow \inf_u \mathcal{E}_{f,\lambda}(u)$, one finds for sufficiently large $k \in \mathbb{N}$ that $\mathcal{E}_{f,\lambda}(u_k) \leq \mathcal{E}_{f,\lambda}(0) = \lambda \|f\|_2^2 < \infty$ up to a subsequence $(u_{\tilde{k}})_{\tilde{k} \in \tilde{\mathbb{N}}}$, thus

3 The Rudin-Osher-Fatemi model

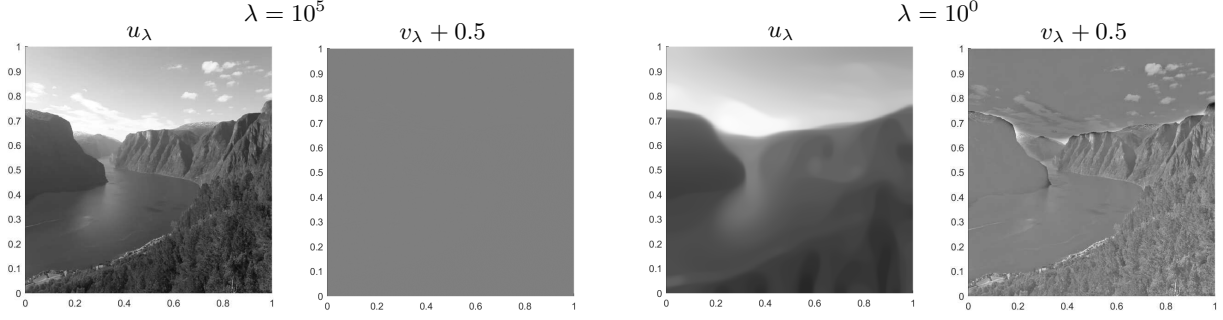


Figure 3.1: ROF decomposition of the *fjord picture* at scale $\lambda = 1e5$ and $\lambda = 1e0$: While with the choice of a “large” λ almost all information of f end up in u_λ , relatively “small” λ yield very smoothed results.

$(u_k)_{k \in \mathbb{N} \setminus \tilde{N}}$ is bounded in $L^2(\Omega)$. Then it follows that – again up to a subsequence – $(u_k)_{k \in \mathbb{N} \setminus \tilde{N}}$ converges weakly to some $u^* \in L^2(\Omega)$ [cf. 4, page 272], i.e.

$$\int_{\Omega} u_k(x) v(x) dx \rightarrow \int_{\Omega} u^*(x) v(x) dx \quad \forall v \in L^2(\Omega).$$

By [4, page 273], it now holds that $\|f - u^*\|_2^2 \leq \liminf_{k \rightarrow \infty} \|f - u_k\|_2^2$. This result in combination with the *lower semi-continuity* of $\|\cdot\|_{BV}$ stated before in Lemma 2.11 a) yields that

$$\mathcal{E}_{f,\lambda}(u^*) \leq \liminf_{k \rightarrow \infty} \mathcal{E}_{f,\lambda}(u_k) = \inf_u \mathcal{E}_{f,\lambda}(u).$$

Hence, u^* is a minimiser of (ROF).

The uniqueness of this minimiser now follows from the convexity of $\mathcal{E}_{f,\lambda}$ which can directly be derived from the convexity of its summands $\|\cdot\|_{BV}$ (see Lemma 2.11 b)) and $\|\cdot\|_2^2$ (follows directly from the triangle inequality), and the following calculation:

Be u' and u'' two minimiser of (ROF), i.e. $\mathcal{E}_{f,\lambda}(u') = \mathcal{E}_{f,\lambda}(u'') = \inf_u \mathcal{E}_{f,\lambda}(u)$. Then:

$$\begin{aligned} \inf_u \mathcal{E}_{f,\lambda}(u) &\leq \mathcal{E}_{f,\lambda}\left(\frac{u' + u''}{2}\right) \\ &\stackrel{\|\cdot\|_{BV} \text{ convex}}{\leq} \frac{1}{2}(\|u'\|_{BV} + \|u''\|_{BV}) + \lambda \left\|f - \frac{u' + u''}{2}\right\|_2^2 \\ &= \frac{1}{2}(\mathcal{E}_{f,\lambda}(u') + \mathcal{E}_{f,\lambda}(u'')) + \lambda \left(\left\|f - \frac{u' + u''}{2}\right\|_2^2 - \frac{1}{2}(\|f - u'\|_2^2 + \|f - u''\|_2^2) \right) \\ &= \frac{1}{2}(\mathcal{E}_{f,\lambda}(u') + \mathcal{E}_{f,\lambda}(u'')) + \lambda \left(\|f\|_2^2 - \langle f, u' + u'' \rangle + \frac{1}{4}\|u'\|_2^2 + \frac{1}{2}\langle u', u'' \rangle + \frac{1}{4}\|u''\|_2^2 \right. \\ &\quad \left. - \frac{1}{2}(\|f\|_2^2 - 2\langle f, u' \rangle + \|u'\|_2^2 + \|f\|_2^2 - 2\langle f, u'' \rangle + \|u''\|_2^2) \right) \\ &= \frac{1}{2}(\mathcal{E}_{f,\lambda}(u') + \mathcal{E}_{f,\lambda}(u'')) + \lambda \left(-\frac{1}{4}\|u'\|_2^2 + \frac{1}{2}\langle u', u'' \rangle - \frac{1}{4}\|u''\|_2^2 \right) \\ &= \frac{1}{2}(\mathcal{E}_{f,\lambda}(u') + \mathcal{E}_{f,\lambda}(u'')) - \frac{\lambda}{4}\|u' - u''\|_2^2 \\ &= \inf_u \mathcal{E}_{f,\lambda}(u) - \frac{\lambda}{4}\|u' - u''\|_2^2 \end{aligned}$$

which would be strictly less than $\inf_u \mathcal{E}_{f,\lambda}(u)$, unless $u' = u''$ almost everywhere. In conclusion:

Theorem 3.1. [cf. 4, Section 1.2.3] The energy functional $\mathcal{E}_{f,\lambda}$ of the minimisation problem (ROF) is *lower semi-continuous* and *strictly convex*. Hence, it has a minimiser and this minimiser is unique.

The convexity of $\mathcal{E}_{f,\lambda}$ is not only important for the uniqueness of a minimiser. It also assures that – at least in a continuous setting – there are no local minima which can interrupt the minimisation process inevitably. Furthermore, this enables one to use many tools from convex optimisation [cf. 4, page 268].

After knowing that a unique minimiser u_λ of (ROF) exists, the question for its properties arises. Of particular interest here are statements on its character regarding the representation of information that the signal f contained. In Chapter 2, I already discussed some aspects on that when describing functions of bounded variation and the impact of a possibly small total variation in detail.

But there is another – clearly desirable – result directly regarding the *ROF model* that I would like to mention: In [4, Theorem 2.7], the authors clarify that at least for an input signal $f \in BV(\Omega) \cap L^\infty(\Omega)$, the *ROF model* does not produce new discontinuities in the sense that the “jump set” of the resultant u_λ is contained in the one of f . Later in [4, page 295], it is further stated that for almost all points in the jump set of u_λ , the orientation of the jumps remains the same as in f .

Now, what can be said about the nature of the minimiser u_λ and its relation to the residual v_λ ? Before coming to the main theorem from Meyer which tries to give an answer on that, I will state two – in my opinion intuitively clear – results regarding constant displacements of the signal f . From here on, I will consider the domain Ω on which f is defined, to be bounded. This is in addition to the previous assumption of Ω being an open set, and appears reasonable in the context of practically dealing with real images. Now the idea regarding the displacements is as follows:

The addition of a constant signal is associated with no change in total variation, since the latter becomes 0 precisely for a.e.-constant functions as discussed in Lemma 2.7, whereas it may certainly have an impact on the L^2 -norm. Thus, in the decomposition, any difference in mean from 0 will be fully included in u_λ , and accordingly the residual v_λ will always have a mean of 0. Although this statement might seem obvious, I will give a short proof:

Lemma 3.2. *Let $n \in \mathbb{N}$, $\Omega \subset \mathbb{R}^n$ be an open and bounded set, $f \in L^2(\Omega)$ and $f = u_\lambda + v_\lambda$ the optimal ROF decomposition. Then the mean of v_λ in Ω is 0:*

$$\int_{\Omega} v_\lambda(x) dx = 0.$$

Proof. Suppose, $[u_\lambda, v_\lambda] = \arg \min_{u+v=f} J(f, \lambda)$ is the optimal *ROF decomposition* of f where the mean of v_λ is non-zero, i.e. $\bar{v} := \frac{1}{|\Omega|} \int_{\Omega} v_\lambda(x) dx \neq 0$. Let $\mathbf{1} : \Omega \rightarrow \mathbb{R}$ with $\mathbf{1}(x) = 1$ for $x \in \Omega$ be the “one-function” and note that $\langle v_\lambda, \mathbf{1} \rangle = \bar{v}|\Omega|$ and $\|\bar{v}\mathbf{1}\|_2^2 = \bar{v}^2|\Omega|$. Then it can be shown that $[u', v']$ with $u' := u_\lambda + \bar{v}\mathbf{1}$ and $v' := v_\lambda - \bar{v}\mathbf{1}$ is a “better” decomposition in the sense that $\mathcal{E}_{f,\lambda}(u') < \mathcal{E}_{f,\lambda}(u_\lambda)$, in contradiction to the optimality of $[u_\lambda, v_\lambda]$. This follows from:

$$\begin{aligned} \mathcal{E}_{f,\lambda}(u_\lambda + \bar{v}\mathbf{1}) &= \|u_\lambda\|_{BV} + \underbrace{\|\bar{v}\mathbf{1}\|_{BV}}_{=0} + \|v_\lambda - \bar{v}\mathbf{1}\|_2^2 \\ &= \|u_\lambda\|_{BV} + \|v_\lambda\|_2^2 + \|\bar{v}\mathbf{1}\|_2^2 - 2\langle v_\lambda, \bar{v}\mathbf{1} \rangle \\ &= \|u_\lambda\|_{BV} + \|v_\lambda\|_2^2 + \bar{v}^2|\Omega| - 2\bar{v}\langle v_\lambda, \mathbf{1} \rangle \\ &= \|u_\lambda\|_{BV} + \|v_\lambda\|_2^2 - \bar{v}^2|\Omega| \\ &< \|u_\lambda\|_{BV} + \|v_\lambda\|_2^2 = \mathcal{E}_{f,\lambda}(u_\lambda). \end{aligned}$$

Thus, u_λ would not minimise (ROF), so the assumption is wrong and it follows $\bar{v} = 0$. □

Corollary. *As direct conclusion from Lemma 3.2 it follows that $\int_{\Omega} u_\lambda(x) dx = \int_{\Omega} f(x) dx$. Thus, the mean of u_λ coincides with the one of f .*

This gives rise to another result: Let \bar{f} denote the mean value of a signal f . Instead of solving (ROF) directly for f , one could formally also solve it for its mean value adjusted counterpart $\tilde{f} := f - \bar{f}$ and obtain the optimal decomposition of f by adding the mean value to the resultant minimiser \tilde{u}_λ , so in the end $f = (\tilde{u}_\lambda + \tilde{f}) + v_\lambda$. This fact is shown in the next lemma and will be important in the application of Meyer's theorem to bounded domains.

Lemma 3.3. *Let $n \in \mathbb{N}$, $\Omega \subset \mathbb{R}^n$ be an open and bounded set, $f \in L^2(\Omega)$ and $f = u_\lambda + v_\lambda$ the optimal ROF decomposition, i.e. $[u_\lambda, v_\lambda] = \arg \min_{u+v=f} J(f, \lambda)$.*

Then the optimal decomposition for the mean value adjusted signal $\tilde{f} := f - \bar{f}\mathbf{1}$ with $\bar{f} := \frac{1}{|\Omega|} \int_\Omega f(x) dx$ is given by $\tilde{f} = \tilde{u}_\lambda + v_\lambda$ where $\tilde{u}_\lambda = u_\lambda - \bar{f}\mathbf{1}$, meaning $[\tilde{u}_\lambda, v_\lambda] = \arg \min_{u+v=\tilde{f}} J(\tilde{f}, \lambda)$.

Furthermore, for their energy it holds $\mathcal{E}_{\tilde{f}, \lambda}(\tilde{u}_\lambda) = \mathcal{E}_{f, \lambda}(u_\lambda)$.

Proof. At first, I will show the equality of energy. Therefor let $[u_\lambda, v_\lambda] := \arg \min_{u+v=f} J(f, \lambda)$, i.e. u_λ minimises $\mathcal{E}_{f, \lambda}$. Further be $\tilde{u}_\lambda := u_\lambda - \bar{f}\mathbf{1}$. When looking at the energy functional $\mathcal{E}_{\tilde{f}, \lambda}$ of the mean value adjusted signal \tilde{f} in \tilde{u}_λ , it can be obtained that

$$\mathcal{E}_{\tilde{f}, \lambda}(\tilde{u}_\lambda) = \|u_\lambda\|_{BV} + \underbrace{\|\bar{f}\mathbf{1}\|_{BV}}_{=0} + \left\| \tilde{f} - u_\lambda + \bar{f}\mathbf{1} \right\|_2^2 = \|u\|_{BV} + \|f - u_\lambda\|_2^2 = \mathcal{E}_{f, \lambda}(u_\lambda).$$

Now, to obtain that \tilde{u}_λ is the actual (ROF) minimiser of \tilde{f} , it suffices to see that for any (non-a.e.-zero) $h \in BV(\Omega)$ it holds

$$\mathcal{E}_{\tilde{f}, \lambda}(\tilde{u}_\lambda + h) = \|u_\lambda + h\|_{BV} + \|f - u_\lambda - h\|_2^2 = \mathcal{E}_{f, \lambda}(u_\lambda + h) > \mathcal{E}_{f, \lambda}(u_\lambda) = \mathcal{E}_{\tilde{f}, \lambda}(\tilde{u}_\lambda)$$

because u_λ minimises $\mathcal{E}_{f, \lambda}$. Hence, \tilde{u}_λ is the (unique) minimiser of $\mathcal{E}_{\tilde{f}, \lambda}$ and the lemma is proven. \square

With these preliminaries, I will now come to the theorem of Meyer. It uses a *dual norm* of $\|\cdot\|_{BV}$, denoted by $\|\cdot\|_*$. I do not want to go into detail on duality in general or the dual to the *BV space* in particular as that is not in the focus of this thesis. Instead, I will directly quote the statements regarding this matter: Following [16, Sections 1.12 and 1.14], for any function $v : \Omega \rightarrow \mathbb{R}$ with $\int_\Omega v(x) dx = 0$ and a representation $v = \operatorname{div} g$ where $g \in C_c^1(\Omega, \mathbb{R}^n)$, the dual norm $\|v\|_*$ takes the form

$$\|v\|_* = \inf_{\substack{g \in C_c^1(\Omega, \mathbb{R}^n) \\ \operatorname{div} g = v}} \|g\|_{L^\infty(\Omega)} = \sup_{\substack{u \in BV(\Omega) \\ \|u\|_{BV} \neq 0}} \frac{\langle u, v \rangle}{\|u\|_{BV}}. \quad (3.4)$$

Furthermore, by [16, Lemma 3] it holds for arbitrary $v \in L^2(\Omega)$ with $\int_\Omega v(x) dx = 0$ that

$$|\langle u, v \rangle| = \left| \int_\Omega u(x)v(x) dx \right| \leq \|u\|_{BV} \|v\|_* \quad \forall u \in BV(\Omega). \quad (3.5)$$

Now, the following and aforesaid Theorem 3.4 corresponds to [16, Theorem 3 and Lemma 4] and provides information on the quantitative character of the *ROF decomposition* $f = u_\lambda + v_\lambda$. It consists of two parts: The first yields that if the parameter λ is chosen too small (or the image f is “too smooth” for that parameter), then in the optimal *ROF decomposition*, $u_\lambda = 0$ and f remains entirely in the residual v_λ . Intuitively, this follows from $\|f\|_2^2$ being a fixed number, so if it is weighted negligibly small in $\mathcal{E}_{f, \lambda}$, any change in u is penalised stronger by $\|u\|_{BV}$ than the improvement in $\|f - u\|_2^2$ could be.

The second part in my interpretation is a statement on the distribution of information in the *ROF decomposition*: If $\|\cdot\|_*$ is considered to be a measure for texture size [16] [23], and the minimisation of (ROF) is seen as the separation of the “essential features” of an image f from its texture, then the theorem states that the *ROF model* resolves all information “down to scale $1/2\lambda$ ”. That “resolved part”

ends up in u_λ , whereas the “texture below this scale” remains in the residual v_λ . Thus, the theorem is also a result that quantifies the influence of the parameter λ on the decomposition.

Note that the original version from Meyer is stated without requiring that the mean of f is 0 because the entire \mathbb{R}^n is taken as domain Ω . But in preparation of the numerics in Chapter 5, I want to give and proof the following modification of Meyer’s theorem for bounded domains Ω :

Theorem 3.4. *Let $n \in \mathbb{N}$, $\Omega \subset \mathbb{R}^n$ be an open and bounded set, $f \in L^2(\Omega)$ with $\int_\Omega f(x)dx = 0$ and $f = u_\lambda + v_\lambda$ the optimal ROF decomposition for $\lambda > 0$, i.e. $[u_\lambda, v_\lambda] = \arg \min_{u+v=f} J(f, \lambda)$. Then it holds:*

- a) *If $\|f\|_* \leq \frac{1}{2\lambda}$, then $[u_\lambda, v_\lambda] = [0, f]$ (almost everywhere). Roughly speaking, this means that the entire image f is treated by the ROF model as texture.*
- b) *Otherwise if $\|f\|_* > \frac{1}{2\lambda}$, then the optimal decomposition $f = u_\lambda + v_\lambda$ is characterised by the following two conditions:*

$$\|v_\lambda\|_* = \frac{1}{2\lambda} \quad \text{and} \quad \langle u_\lambda, v_\lambda \rangle = \int_\Omega u_\lambda(x)v_\lambda(x)dx = \frac{1}{2\lambda} \|u_\lambda\|_{BV}.$$

Proof. The proof for this theorem is an extension of the proof given in [16, pages 32-34].

- a) For u_λ , no (almost everywhere) constant function other than 0 needs to be considered because by Lemma 3.2 it is $\int_\Omega u_\lambda(x)dx = \int_\Omega f(x)dx = 0$; and a non-zero constant function has non-zero mean, contrary to the setting of the theorem.

Since $u_\lambda = \arg \min_{u \in BV(\Omega)} \mathcal{E}_{f,\lambda}(u)$, the problem (ROF) yields $u_\lambda = 0$ if and only if for any function $h \in BV(\Omega)$ with $\|h\|_{BV} \neq 0$ it holds

$$\begin{aligned} & \mathcal{E}_{f,\lambda}(0) \leq \mathcal{E}_{f,\lambda}(h) \\ \Leftrightarrow & \lambda \|f\|_2^2 \leq \|h\|_{BV}^2 + \lambda \|f - h\|_2^2 \\ \Leftrightarrow & \lambda \|f\|_2^2 \leq \|h\|_{BV}^2 + \lambda \|f\|_2^2 + \lambda \|h\|_2^2 - 2\lambda \langle h, f \rangle \\ \Leftrightarrow & \langle h, f \rangle \leq \frac{1}{2\lambda} \|h\|_{BV}^2 + \frac{1}{2} \|h\|_2^2 \\ \Leftrightarrow & \frac{\langle h, f \rangle}{\|h\|_{BV}} \leq \frac{1}{2\lambda} + \frac{1}{2} \frac{\|h\|_2^2}{\|h\|_{BV}}. \end{aligned}$$

Now replacing h by εh for $\varepsilon > 0$ and letting $\varepsilon \rightarrow 0$ yields:

$$\begin{aligned} & \frac{\langle \varepsilon h, f \rangle}{\|\varepsilon h\|_{BV}} \leq \frac{1}{2\lambda} + \frac{1}{2} \frac{\|\varepsilon h\|_2^2}{\|\varepsilon h\|_{BV}} \\ \Leftrightarrow & \frac{|\varepsilon| \langle h, f \rangle}{|\varepsilon| \|h\|_{BV}} \leq \frac{1}{2\lambda} + \frac{1}{2} \frac{|\varepsilon|^2 \|h\|_2^2}{|\varepsilon| \|h\|_{BV}} \\ \Leftrightarrow & \frac{\langle h, f \rangle}{\|h\|_{BV}} \leq \frac{1}{2\lambda} + \frac{|\varepsilon|}{2} \frac{\|h\|_2^2}{\|h\|_{BV}} \xrightarrow{\varepsilon \rightarrow 0} \frac{1}{2\lambda}. \end{aligned}$$

But with (3.4), it is

$$\|f\|_* = \sup_{\substack{h \in BV(\Omega) \\ \|h\|_{BV} \neq 0}} \frac{\langle h, f \rangle}{\|h\|_{BV}} \leq \frac{1}{2\lambda}.$$

Hence, if $\|f\|_* \leq \frac{1}{2\lambda}$, then $u_\lambda = 0$ almost everywhere and statement a) follows.

b) Now be $\|f\|_* > \frac{1}{2\lambda}$. Since $[u_\lambda, v_\lambda] = \arg \min_{u+v=f} J(f, \lambda)$ minimises (ROF) and $\|\cdot\|_{BV}$ is convex, it holds for any $h \in BV(\Omega)$ and $\varepsilon \in \mathbb{R}$ that

$$\begin{aligned} \|u_\lambda\|_{BV} + \lambda \|v_\lambda\|_2^2 &\leq \|u_\lambda + \varepsilon h\|_{BV} + \lambda \|v_\lambda - \varepsilon h\|_2^2 \\ &\leq \|u_\lambda\|_{BV} + \|\varepsilon h\|_{BV} + \lambda \|v_\lambda - \varepsilon h\|_2^2. \end{aligned} \quad (3.6)$$

$$\Rightarrow \quad \lambda \|v_\lambda\|_2^2 \leq \|\varepsilon h\|_{BV} + \lambda \|v_\lambda - \varepsilon h\|_2^2.$$

When assuming $\varepsilon > 0$ and $\|h\|_{BV} \neq 0$, the same calculations as in the prove of a) yield:

$$\begin{aligned} \lambda \|v_\lambda\|_2^2 &\leq \|\varepsilon h\|_{BV} + \lambda \|v_\lambda - \varepsilon h\|_2^2 \\ \Leftrightarrow \quad \frac{\langle h, v_\lambda \rangle}{\|h\|_{BV}} &\leq \frac{1}{2\lambda} + \frac{\varepsilon}{2} \frac{\|h\|_2^2}{\|h\|_{BV}} \xrightarrow{\varepsilon \rightarrow 0} \frac{1}{2\lambda}. \end{aligned}$$

And again with (3.4), it is

$$\|v_\lambda\|_* = \sup_{\substack{h \in BV(\Omega) \\ \|h\|_{BV} \neq 0}} \frac{\langle h, v_\lambda \rangle}{\|h\|_{BV}} \leq \frac{1}{2\lambda}. \quad (3.7)$$

Next, taking $h = u_\lambda$ in eq. (3.6) and limiting $\varepsilon \in (-1, 1)$ gives:

$$\begin{aligned} \|u_\lambda\|_{BV} + \lambda \|v_\lambda\|_2^2 &\leq (1 + \varepsilon) \|u_\lambda\|_{BV} + \lambda \|v_\lambda - \varepsilon u_\lambda\|_2^2 \\ &= \|u_\lambda\|_{BV} + \varepsilon \|u_\lambda\|_{BV} + \lambda \|v_\lambda\|_2^2 + \varepsilon^2 \|u_\lambda\|_2^2 - 2\lambda \varepsilon \langle u_\lambda, v_\lambda \rangle \\ \Leftrightarrow \quad 2\lambda \varepsilon \langle u_\lambda, v_\lambda \rangle &\leq \varepsilon \|u_\lambda\|_{BV} + \varepsilon^2 \|u_\lambda\|_2^2 \end{aligned}$$

This inequality holds for arbitrary $\varepsilon \in (-1, 1)$; thus for $\varepsilon \searrow 0$ (i.e. $\varepsilon > 0$) it yields $2\lambda \langle u_\lambda, v_\lambda \rangle \leq \|u_\lambda\|_{BV}$, and for $\varepsilon \nearrow 0$ (i.e. $\varepsilon < 0$) it yields $2\lambda \langle u_\lambda, v_\lambda \rangle \geq \|u_\lambda\|_{BV}$. Together:

$$\langle u_\lambda, v_\lambda \rangle = \frac{1}{2\lambda} \|u_\lambda\|_{BV}.$$

So the second statement of (b) is proven. At last, by (3.5) it holds

$$\frac{1}{2\lambda} \|u_\lambda\|_{BV} = \langle u_\lambda, v_\lambda \rangle \leq \|u_\lambda\|_{BV} \|v_\lambda\|_* \quad (3.8)$$

which yields $\|v_\lambda\|_* \geq \frac{1}{2\lambda}$ since $\|u_\lambda\|_{BV} \neq 0$. The latter must hold because $u_\lambda = 0$ would result in case a) and other constant functions are excluded by Lemma 3.2. Now, (3.8) together with (3.7) finally gives $\|v_\lambda\|_* = \frac{1}{2\lambda}$ which completes the proof. \square

Remark. For arbitrary $f \in L^2(\Omega)$ (with possibly non-zero mean), Lemma 3.3 gives a formal way to apply this theorem. Then, $\|f - \bar{f}\mathbf{1}\|_*$ is evaluated, $[u_\lambda, v_\lambda] = [\bar{f}\mathbf{1}, f - \bar{f}\mathbf{1}]$ is obtained in case a), and in case b) the last equality holds for $u_\lambda - \bar{f}\mathbf{1}$ instead of u_λ .

Note that Theorem 3.4 implies that the inequality (3.5) becomes an equality when the pair $[u, v]$ is the minimising *ROF decomposition* $f = u + v$. This gave rise to Meyer's proposal of a modification of the *ROF model* in which the residual v_λ is evaluated by $\|v_\lambda\|_*$ instead of $\|v_\lambda\|_2^2$ [cf. 16]. To this approach also discrete algorithms were developed, e.g. in [23].

3.3 The Euler-Lagrange equation of the ROF functional

To obtain the solution u_λ of (ROF), the convex functional $\mathcal{E}_{f,\lambda}$ needs to be minimised. From *calculus of variation* it is known that – under some assumptions – the associated *Euler-Lagrange equation* is a necessary condition for a function u to be a minimiser of that functional. For convex functionals with a unique minimiser such as $\mathcal{E}_{f,\lambda}$, satisfying that equation is furthermore sufficient [cf. 19, page 47]. Thus, a possible approach to compute the (ROF) minimiser arises from trying to find a solution of the *Euler-Lagrange equation*. This idea is widely used to tackle (ROF) – amongst others in the original paper [20] and the one proposing the hierarchical decomposition [22] – and will also be the basis for the algorithms used and discussed later in Chapter 5.

Following [19, pages 48-50], the *Euler-Lagrange equation* for a functional $\mathcal{F} : W^{1,p}(\Omega) \rightarrow \mathbb{R}$ of the form $\mathcal{F}(u) = \int_\Omega \mathcal{L}(x, u(x), Du(x)) dx$ is generally given by

$$0 \stackrel{!}{=} \frac{\partial \mathcal{L}}{\partial u} - \operatorname{div} \left(\left(\frac{\partial \mathcal{L}}{\partial u_{x_1}}, \dots, \frac{\partial \mathcal{L}}{\partial u_{x_n}} \right) \right) = \frac{\partial \mathcal{L}}{\partial u} - \left(\frac{\partial}{\partial x_1} \frac{\partial \mathcal{L}}{\partial u_{x_1}} + \dots + \frac{\partial}{\partial x_n} \frac{\partial \mathcal{L}}{\partial u_{x_n}} \right) \quad \text{a.e. in } \Omega, \quad (3.9)$$

augmented by a boundary condition on $\partial\Omega$. The requirement $u \in W^{1,p}(\Omega)$ demands that its weak gradient $Du = (u_{x_1}, \dots, u_{x_n})$ exists and is L^p -integrable. This is clearly a limitation and will not be met in general. However, Lemma 2.12 a) at least ensures that $u_\lambda \in BV(\Omega)$ can be approximated by an L^p -convergent sequence of smooth functions, and hence when computing an approximation to u_λ in a discrete setting, it appears reasonable to make use of the equation nevertheless.

If $u \in BV(\Omega)$ is weakly differentiable, Lemma 2.6 states that the total variation of u is given by $\|u\|_{BV} = \int_\Omega |Du(x)| dx$. Thus, the functional $\mathcal{E}_{f,\lambda}$ takes the form

$$\mathcal{E}_{f,\lambda}(u) = \|u\|_{BV} + \lambda \|f - u\|_2^2 = \int_\Omega \left(\sqrt{u_{x_1}(x)^2 + \dots + u_{x_n}(x)^2} + \lambda (f(x) - u(x))^2 \right) dx. \quad (3.10)$$

Plugging (3.10) into eq. (3.9) and rearranging the terms, finally gives the following *Euler-Lagrange equation* for $\mathcal{E}_{f,\lambda}(u)$ (see also [4, page 284] or [22, page 565]):

$$0 \stackrel{!}{=} u(x) - f(x) - \frac{1}{2\lambda} \operatorname{div} \left(\frac{Du(x)}{|Du(x)|} \right) \quad \forall x \in \Omega. \quad (3.11)$$

with the *Neumann boundary condition*

$$Du \cdot n = 0 \quad \text{on } \partial\Omega. \quad (3.12)$$

The latter imposes that the weak directional derivative of u along the normal n of the boundary $\partial\Omega$ is zero, meaning that there is no flux inwards or outwards the domain Ω . For signals such as images, this condition certainly makes sense. An alternative notation is $\frac{\partial u}{\partial n} \Big|_{\partial\Omega} = 0$.

Besides this requirement of $u_\lambda \in W^{1,p}(\Omega)$, the *Euler-Lagrange equation* for the *ROF model* yields another, possibly even greater issue: Du is only differentiable where $Du(x) \neq 0$. The *ROF model*, however, tends to result in “staircasing”, i.e. a function u_λ with large areas where $Du_\lambda = 0$ [cf. 4, page 284]. Numerically, this difficulty can for instance be met by employing the regularisation $\sqrt{\varepsilon^2 + |Du|^2}$ instead of $|Du|$ in the denominator [22]. I will give some more on that notion in Section 5.2.

4

Extension of the ROF model to a hierarchical (BV, L^2) decomposition

4.1 Description of the hierarchical decomposition

So far, only the single *ROF decomposition* was discussed. In that, the information of an image which can properly be described and represented in the BV space – particularly edges and large homogeneous regions – are being separated from the rest, for instance from noise and texture. But according to the authors of [22], many information and hence images themselves lie somewhere “between” the larger L^2 space and its (smaller) subspace BV ; they are elements of so-called *intermediate spaces*.

From that fact arose the idea to not ultimately decompose an image into two fixed parts, but rather seek a representation in which the various elements are contained in multiple different instances. In view of this, Tadmor, Nezzar and Vese proposed a multiscale representation in 2004 [22]. In this, a given image is hierarchically decomposed by iteratively solving (ROF). Quoting [22, page 556], by doing so, the “intermediate regularity between L^2 and BV ” can be captured and the “representation of an L^2 image is not predetermined but is resolved in terms of layers of intermediate scales”.

In the following, I will introduce that image representation on the basis of [22]: The idea behind the hierarchical decomposition is that information which at a given scale λ are interpreted by the *ROF model* as “textures”, meaning which remained in the residual v_λ , might be seen as “significant edges” when viewed under a larger scale, for instance 2λ . Hence, every residual from the previous decomposition $f = u_\lambda + v_\lambda$ might partly consist of scale-dependent BV features that can be extracted at a refined next step

$$v_\lambda = u_{2\lambda} + v_{2\lambda}, \quad [u_{2\lambda}, v_{2\lambda}] = \arg \min_{u+v=v_\lambda} J(v_\lambda, 2\lambda). \quad (4.1)$$

Now, a two-scale representation $f = u_\lambda + u_{2\lambda} + v_{2\lambda}$ is obtained where u_λ and $u_{2\lambda}$ represent BV -features at different scales, and textures below the scale $1/2\lambda$ remain unresolved in v_λ . This concept of course can be extended to an iterative procedure:

In a first step, for a given image $f \in L^2(\Omega)$ a very coarse decomposition

$$f = u_0 + v_0, \quad [u_0, v_0] = \arg \min_{u+v=f} J(f, \lambda_0) \quad (4.2)$$

is computed in which the resultant minimiser u_0 covers only the “most basic” features of f , while great parts of the image will be interpreted as texture and thus end up in the residual v_0 . This is achieved by choosing a “relatively small” parameter $\lambda_0 > 0$. Next, with the refined scale $\lambda_1 := 2\lambda_0$ that residual is again decomposed to

$$v_0 = u_1 + v_1, \quad [u_1, v_1] = \arg \min_{u+v=v_0} J(v_0, \lambda_1).$$

This approach is continued with the successive application of the refinement step (4.1): In every iteration, the current residual is decomposed at a scale twice the size of the previous one – up to a final

level $m \in \mathbb{N}$. In conclusion, the entire procedure can be described by the initialisation (4.2) followed by

$$v_{\ell-1} = u_\ell + v_\ell, \quad [u_\ell, v_\ell] = \arg \min_{u+v=v_{\ell-1}} J(v_{\ell-1}, 2^\ell \lambda_0) \quad (4.3)$$

for $\ell = 1, \dots, m$. After m steps, this results in the following hierarchical decomposition:

$$\begin{aligned} f &= u_0 + v_0 \\ &= u_0 + u_1 + v_1 \\ &\vdots \\ &= u_0 + u_1 + \dots + u_m + v_m. \end{aligned}$$

Thus, the original image f is up to scale $1/2\lambda_m$ with $\lambda_m := 2^m \lambda_0$ approximated by the multiscale representation of bounded variation

$$f \sim \sum_{\ell=0}^m u_\ell \quad (4.4)$$

with residual v_m , in which each u_ℓ captures certain information of f at the fixed scale $\lambda_\ell := 2^\ell \lambda_0$. Here, I use the “ \sim notation” of [22] and assume that the authors intend to describe precisely this “approximation up to some scale” relation by it.

As m increases, more and more edges are resolved by u_m and the representation of f gets more accurate: $\|v_m\|_2^2$ decreases monotonously with $\|v_m\|_2^2 - \|v_{m+1}\|_2^2 = \frac{1}{\lambda_{m+1}} \|u_{m+1}\|_{BV} + \|u_{m+1}\|_2^2 \geq 0$ (see the proof of Theorem 4.1). Of course, the refinement scale needs not be fixed to factor 2 but may be chosen differently by using $\lambda_\ell = s^\ell \lambda_0$ for any $s > 1$ – the greater s , the coarser the decomposition of f .

The authors of [22] state that the sum $f \sim \sum_{\ell=0}^m u_\ell$ “provides a multilayered description of f ” which lies precisely in the “intermediate scale of spaces, in between BV and L^2 ”, and that this multiscale (BV, L^2) expansion “is particularly suitable for image representations” [22, page 557] which after all considerations in the previous sections, appears very reasonable to me.

In Section 3.1, I mentioned that the choice of λ in the standard *ROF model* needs to be made wisely in order to obtain a proper decomposition $f = u + v$. In case of the hierarchical (BV, L^2) decomposition, this problem is now mostly evaded as the information of f are contained in multiple layers of different scales. The only greater concern regarding this matter is the question how to choose λ_0 properly, even though – except for a shift in the index ℓ – this choice influences the resultant representation of f only slightly. That question of initialisation will be discussed in Section 4.3 where I propose a new method inspired by [22].

4.2 Convergence of the hierarchical decomposition

Naturally, now the question arises how well (4.4) approximates the signal f , in particular whether the sum converges to f and if so, by what rate this convergence can be quantified. In [22], Tadmor, Nezzar and Vese provide mainly two results to these questions which I want to quote next including their proofs.

But before that, in my opinion the following thought [cf. 22, page 558] already gives an idea of what one might expect: When comparing the energy $\mathcal{E}_{v_\ell, \lambda_{\ell+1}}$ of the optimal decomposition $v_\ell = u_{\ell+1} + v_{\ell+1}$ of step ℓ – that is $[u_{\ell+1}, v_{\ell+1}] = \arg \min_{u+v=v_\ell} J(v_\ell, \lambda_{\ell+1})$ – with the one of the trivial pair $[0, v_\ell]$, one finds

$$\begin{aligned} \mathcal{E}_{v_\ell, \lambda_{\ell+1}}(u_\ell) \leq \mathcal{E}_{v_\ell, \lambda_{\ell+1}}(0) &\Leftrightarrow \|u_{\ell+1}\|_{BV} + \lambda_{\ell+1} \|v_{\ell+1}\|_2^2 \leq \lambda_{\ell+1} \|v_\ell\|_2^2 \\ &\Leftrightarrow \frac{1}{\lambda_{\ell+1}} \|u_{\ell+1}\|_{BV} \leq \|v_\ell\|_2^2 - \|v_{\ell+1}\|_2^2, \end{aligned}$$

so when summing over all iterations and using $f = u_0 + v_0$, it holds

$$\begin{aligned} \sum_{\ell=0}^m \frac{1}{\lambda_\ell} \|u_\ell\|_{BV} &= \frac{1}{\lambda_0} \|u_0\|_{BV} + \sum_{\ell=0}^{m-1} \frac{1}{\lambda_{\ell+1}} \|u_{\ell+1}\|_{BV} \\ &\leq \|f\|_2^2 - \|v_0\|_2^2 + \sum_{\ell=0}^{m-1} \left(\|v_\ell\|_2^2 - \|v_{\ell+1}\|_2^2 \right) \\ &= \|f\|_2^2 - \|v_m\|_2^2 \\ &\leq \|f\|_2^2. \end{aligned}$$

Thus, when weighting the total variation of each instance u_ℓ with its corresponding scale $1/\lambda_\ell$ and summing them all up, this sum is bounded by the “ L^2 -energy” of f for any m . The next theorem gives a more precise “energy estimate” [22, page 558] as well as a statement on the convergence of the hierarchical decomposition $f \sim \sum_\ell u_\ell$:

Theorem 4.1. [cf. 22, Theorem 2.1] *Let $n \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set and $f \in L^2(\Omega)$. Then f admits the hierarchical decomposition*

$$f = \sum_{\ell=0}^{\infty} u_\ell \quad \text{with the “(weak) convergence rate”} \quad \left\| f - \sum_{\ell=0}^m u_\ell \right\|_* \leq \frac{1}{2^{m+1}\lambda_0}. \quad (4.5)$$

Furthermore, the following “energy” estimate holds:

$$\sum_{\ell=0}^{\infty} \left(\frac{1}{\lambda_\ell} \|u_\ell\|_{BV} + \|u_\ell\|_2^2 \right) \leq \|f\|_2^2, \quad \lambda_\ell := 2^\ell \lambda_0. \quad (4.6)$$

Proof. The proof follows [22] and is mainly based on Theorem 3.4, given by Meyer in 2001. Since in my version it requires $\int_\Omega f(x)dx = 0$, I will first proof the statement under this assumption: According to that theorem, for the (ROF) decomposition $f = u_\lambda + v_\lambda$ it holds: If $\|f\|_* \leq \frac{1}{2\lambda}$, then $[u_\lambda, v_\lambda] = [0, f]$; otherwise,

$$\|v_\lambda\|_* = \frac{1}{2\lambda} \quad \text{and} \quad \langle u_\lambda, v_\lambda \rangle = \frac{1}{2\lambda} \|u_\lambda\|_{BV}. \quad (4.7)$$

For $m \in \mathbb{N}$, $[u_m, v_m]$ is given by $[u_m, v_m] = \arg \min_{u+v=v_{m-1}} J(v_{m-1}, \lambda_m)$, and Lemma 3.2 assures that v_{m-1} has mean 0, so the theorem still holds. Now either:

- $\|v_{m-1}\|_* \leq \frac{1}{2\lambda_m}$, so $[u_m, v_m] = [0, v_{m-1}]$ and hence $\|v_m\|_* = \|v_{m-1}\|_* \leq \frac{1}{2\lambda_m}$,
- or if $\|v_{m-1}\|_* > \frac{1}{2\lambda_m}$, then by (4.7) it holds $\|v_m\|_* = \frac{1}{2\lambda_m}$.

Together:

$$\left\| f - \sum_{\ell=0}^m u_\ell \right\|_* = \|v_m\|_* \leq \frac{1}{2\lambda_m} = \frac{1}{2^{m+1}\lambda_0},$$

so (4.5) is shown. For the second statement, squaring the basic refinement step $v_\ell = u_{\ell+1} + v_{\ell+1}$ and using (4.7) gives:

$$\begin{aligned} \|v_\ell\|_2^2 &= \|u_{\ell+1} + v_{\ell+1}\|_2^2 \\ &= \|u_{\ell+1}\|_2^2 + \|v_{\ell+1}\|_2^2 + 2\langle u_{\ell+1}, v_{\ell+1} \rangle \\ &= \|u_{\ell+1}\|_2^2 + \|v_{\ell+1}\|_2^2 + \frac{1}{\lambda_{\ell+1}} \|u_{\ell+1}\|_{BV} \end{aligned}$$

which is equivalent to $\frac{1}{\lambda_{\ell+1}} \|u_{\ell+1}\|_{BV} + \|u_{\ell+1}\|_2^2 = \|v_\ell\|_2^2 - \|v_{\ell+1}\|_2^2$ for $\ell = 0, \dots, m-1$. In the first iteration it analogously holds $\frac{1}{\lambda_0} \|u_0\|_{BV} + \|u_0\|_2^2 = \|f\|_2^2 - \|v_0\|_2^2$.

Summing this up finally yields for all $m \in \mathbb{N}$:

$$\sum_{\ell=0}^m \left(\frac{1}{\lambda_\ell} \|u_\ell\|_{BV} + \|u_\ell\|_2^2 \right) = \|f\|_2^2 - \|v_0\|_2^2 + \sum_{\ell=1}^m \left(\|v_{\ell-1}\|_2^2 - \|v_\ell\|_2^2 \right) = \|f\|_2^2 - \|v_m\|_2^2 \leq \|f\|_2^2 \quad (4.8)$$

Thus, also the energy estimate (4.6) is shown.

Now to the case that Ω is bounded and $\bar{f} := \frac{1}{|\Omega|} \int_\Omega f(x) dx \neq 0$: Lemma 3.3 allows the application of Theorem 3.4 and thus the above considerations for $\tilde{f} := f - \bar{f}\mathbf{1}$. Since then $u_0 = \tilde{u}_0 + \bar{f}\mathbf{1}$, the statement (4.5) still holds for f .

To check the validity of (4.6), further note that \tilde{f} and \tilde{u}_0 have mean 0, so $\langle \tilde{u}_0, \mathbf{1} \rangle = \langle \tilde{f}, \mathbf{1} \rangle = 0$. Thus,

$$\begin{aligned} \|f\|_2^2 &= \|\tilde{f} + \bar{f}\mathbf{1}\|_2^2 = \|\tilde{f}\|_2^2 + \|\bar{f}\mathbf{1}\|_2^2 + \underbrace{2\bar{f}\langle \tilde{f}, \mathbf{1} \rangle}_{=0} \\ \|u_0\|_2^2 &= \|\tilde{u}_0 + \bar{f}\mathbf{1}\|_2^2 = \|\tilde{u}_0\|_2^2 + \|\bar{f}\mathbf{1}\|_2^2 + \underbrace{2\bar{f}\langle \tilde{u}_0, \mathbf{1} \rangle}_{=0} \end{aligned}$$

Since also $\|u_0\|_{BV} = \|\tilde{u}_0\|_{BV} + \|\bar{f}\mathbf{1}\|_{BV} = \|\tilde{u}_0\|_{BV}$, (4.6) holds for arbitrary $f \in BV(\Omega)$ as well and the prove is complete. \square

Note that the original statement of [22] uses

$$\|f\|_{W^{-1,\infty}} := \sup_{\substack{g \in W^{1,1}(\Omega) \\ \|\nabla g\|_{L^1} \neq 0}} \frac{\int_\Omega f(x)g(x)dx}{\|\nabla g\|_{L^1}} \quad \text{instead of} \quad \|f\|_* = \sup_{\substack{g \in BV(\Omega) \\ \|g\|_{BV} \neq 0}} \frac{\int_\Omega f(x)g(x)dx}{\|g\|_{BV}}$$

which – in a continuously differentiable setting – is identical by Theorem 2.4. Furthermore, the ‘‘convergence rate’’ in (4.5) that I wrote as an inequality, is given as an equality there, which I do not agree with in general: As can be seen from the proof, this would require $v_m \neq 0$ for all $m \in \mathbb{N}$ which does not necessarily hold (e.g. if f is constant, then the residual will always be 0).

Due to the authors of [22], (4.5) is limited to *weak convergence*. However, they give a second result yielding *strong convergence* if $f \in BV(\Omega)$. At the same time, also the energy inequality (4.6) becomes an equality because by eq. (4.8), this is the case precisely when $\|v_m\|_2^2 = \|f - \sum_{\ell=0}^m u_\ell\|_2^2 \rightarrow 0$ as m goes to infinity – strong convergence in L^2 is attained:

Theorem 4.2. [cf. 22, Theorem 2.2] *Let $n \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set and $f \in BV(\Omega)$. Then the hierarchical (BV, L^2) decomposition $f = \sum_{\ell=0}^\infty u_\ell$ converges strongly in L^2 , and the energy of f is given by*

$$\sum_{\ell=0}^\infty \left(\frac{1}{\lambda_\ell} \|u_\ell\|_{BV} + \|u_\ell\|_2^2 \right) = \|f\|_2^2, \quad \lambda_\ell := 2^\ell \lambda_0. \quad (4.9)$$

Proof. To proof the theorem, it must be shown that $\|v_m\|_2^2 \rightarrow 0$ or equivalently $\|v_{2m}\|_2^2 \rightarrow 0$ as $m \rightarrow \infty$. This residual v_{2m} can be written as $v_{2m} = v_m - \sum_{\ell=m+1}^{2m} u_\ell$, and multiplying it against v_{2m} yields:

$$\|v_{2m}\|_2^2 = \langle v_{2m}, v_{2m} \rangle = - \underbrace{\left\langle v_{2m}, \sum_{\ell=m+1}^{2m} u_\ell \right\rangle}_{=:I} + \underbrace{\langle v_{2m}, v_m \rangle}_{=:II} \leq |I| + |II|.$$

Hence, when showing that $|I|$ and $|II|$ tend to zero as $m \rightarrow \infty$, the proof is complete. From the previous proof it is known that $\|v_{2m}\|_* \leq 1/2\lambda_{2m}$, so for arbitrary $h \in BV(\Omega)$ it holds by (3.5) that $|\langle v_{2m}, h \rangle| \leq \|v_{2m}\|_* \|h\|_{BV} \leq \frac{1}{2\lambda_{2m}} \|h\|_{BV}$. Now, for I follows:

$$|I| \leq \frac{1}{2\lambda_{2m}} \left\| \sum_{\ell=m+1}^{2m} u_\ell \right\|_{BV} \leq \frac{1}{2\lambda_{2m}} \sum_{\ell=m+1}^{2m} \|u_\ell\|_{BV} \leq \sum_{\ell=m+1}^{2m} \frac{1}{2\lambda_\ell} \|u_\ell\|_{BV} \xrightarrow{m \rightarrow \infty} 0$$

as a Cauchy subsequence of the bounded series $\sum_{\ell=0}^{\infty} \left(\frac{1}{\lambda_\ell} \|u_\ell\|_{BV} + \|u_\ell\|_2^2 \right) \leq \|f\|_2^2$, given in (4.6).

For II , the requirement $f \in BV(\Omega)$ is now needed: Since f and all u_ℓ have bounded variation, this is also true for the residual v_m . Furthermore, the growth of $\|v_m\|_{BV}$ is limited by 2^m ; in particular:

$$\|v_m\|_{BV} = \left\| f - \sum_{\ell=0}^m u_\ell \right\|_{BV} \leq \|f\|_{BV} + \sum_{\ell=0}^m \|u_\ell\|_{BV} \leq \|f\|_{BV} + \lambda_m \sum_{\ell=0}^m \frac{1}{\lambda_\ell} \|u_\ell\|_{BV} \leq \|f\|_{BV} + \lambda_m \|f\|_2^2$$

Finally, with $|\langle v_{2m}, v_m \rangle| \leq \frac{1}{2\lambda_{2m}} \|v_m\|_{BV}$ (by the same argument as above), $\|f\|_{BV} < \infty$ and $\|f\|_2^2 < \infty$, it can be concluded that $|II|$ vanishes:

$$|II| \leq \frac{1}{2\lambda_{2m}} \|v_m\|_{BV} \leq \frac{1}{2\lambda_{2m}} \left(\|f\|_{BV} + \lambda_m \|f\|_2^2 \right) = \frac{1}{2^{2m+1}\lambda_0} \|f\|_{BV} + \frac{1}{2^{m+1}} \|f\|_2^2 \xrightarrow{m \rightarrow \infty} 0.$$

□

The authors of [22] even extend the statement of strong convergence and equality in the energy estimate to functions from the intermediate space of L^2 and BV , and conclude that other extensions in the same manner were possible: “A minimal amount of smoothness beyond the L^2 bound will guarantee strong convergence” [22, page 559]. This guarantees that under very little assumptions on a given image f , any information of it will at one point be resolved by the hierarchical decomposition and thus contained in the resultant multiscale representation $f \sim \sum_\ell u_\ell$.

Finally, in [22] is stated that the decomposition of energy stated in (4.9) lies entirely within the BV scales [cf. 22, page 560]. This limits the variation of the latest resolved BV features u_ℓ in each step ℓ and stresses the relation between the energy of f and the character of its multiscale representation. The statement requires strong convergence of the hierarchical decomposition, for instance by $f \in BV(\Omega)$, and that $\int_\Omega f(x) dx = 0$ – otherwise, of course it would apply to the mean value adjusted signal:

Lemma 4.3. *Let $m \in \mathbb{N}$, $\Omega \subseteq \mathbb{R}^n$ be an open set, $f \in L^2(\Omega)$ and $\int_\Omega f(x) dx = 0$. If the energy of f is given by $\|f\|_2^2 = \sum_{\ell=0}^{\infty} \left(\frac{1}{\lambda_\ell} \|u_\ell\|_{BV} + \|u_\ell\|_2^2 \right)$ as in (4.9), then it lies entirely in the BV scales with the following boundaries:*

$$\sum_{\ell=0}^{\infty} \frac{1}{\lambda_\ell} \|u_\ell\|_{BV} \leq \|f\|_2^2 \leq \frac{5}{2} \sum_{\ell=0}^{\infty} \frac{1}{\lambda_\ell} \|u_\ell\|_{BV}.$$

Proof. The first inequality holds by definition, so I will only address the second: Since $u_\ell = v_{\ell-1} - v_\ell$, it follows

$$\|u_\ell\|_* \leq \|v_{\ell-1}\|_* + \|v_\ell\|_* \leq \frac{1}{2\lambda_{\ell-1}} + \frac{1}{2\lambda_\ell} = \frac{3}{2\lambda_\ell}.$$

Furthermore, again by (3.5) it is $\|u_\ell\|_2^2 = |\langle u_\ell, u_\ell \rangle| \leq \|u_\ell\|_{BV} \|u_\ell\|_*$ and hence $\|u_\ell\|_2^2 \leq \frac{3}{2\lambda_\ell} \|u_\ell\|_{BV}$. Together, the final result is obtained:

$$\sum_{\ell=0}^{\infty} \frac{1}{\lambda_\ell} \|u_\ell\|_{BV} \leq \underbrace{\sum_{\ell=0}^{\infty} \left(\frac{1}{\lambda_\ell} \|u_\ell\|_{BV} + \|u_\ell\|_2^2 \right)}_{=\|f\|_2^2} \leq \frac{5}{2} \sum_{\ell=0}^{\infty} \frac{1}{\lambda_\ell} \|u_\ell\|_{BV}.$$

□

Note: In the original version of [22], the upper bound is given by $\frac{3}{2} \sum_{\ell=0}^{\infty} \frac{1}{\lambda_\ell} \|u_\ell\|_{BV}$. I do not follow on that result, but the overall interpretation, however, remains the same in both cases.

4.3 Initialisation of the hierarchical decomposition

At last, I will address the question how to choose the scale for the first step, namely λ_0 . As discussed in Chapter 3, a “small” weighting parameter $\lambda > 0$ in (ROF) yields a decomposition in which relatively few elements are extracted in u_λ . And by Theorem 3.4 a) it is known that – besides a mean displacement – u_λ is empty if $\|f\|_* \leq 1/2\lambda$, that is $\lambda \leq 1/2\|f\|_*$.

The procedure (4.3) mainly aims at decomposing the signal f in a way that its major parts are distributed to several different instances (with corresponding scale). Thus, λ_0 should be chosen such that in the first result u_0 not many (or rather the least possible amount of) information are being captured and every following u_ℓ significantly resolves new edges. On the other hand, if λ_0 is chosen too small (way below $1/2\|f\|_*$), then the procedure will yield $u_\ell = 0$ until the parameter $\lambda_\ell = 2^\ell \lambda_0$ exceeds $1/2\|f\|_*$.

So, with scaling factor 2, the initial scale λ_0 is sought such that

$$\frac{1}{2\lambda_0} \leq \|f\|_* \leq \frac{1}{\lambda_0}.$$

In practice, it might be impossible to compute (or even properly estimate) $\|f\|_*$. Therefore, the authors of [22] propose a reverse refinement procedure which “aims to capture a hierarchical representation of the missing larger scales.” However, I am convinced that their method of decomposing the residual v_0 downwards, cannot work because $\|v_0\|_* = \frac{1}{2\lambda_0} < \frac{1}{\lambda_0} = \frac{1}{2\lambda_{-1}}$ yields $[0, v_0]$ by Theorem 3.4 a) (for details on their procedure, see [22, page 561]). Thus, inspired by their concept, I propose a different approach.

The idea is simple: If λ_0 was chosen too large, then u_0 contains too many information and should be decomposed downwards (i.e. with a smaller scale $\lambda_{-1} := 2^{-1}\lambda_0$). To still obtain a representation $f \sim \sum_\ell u_\ell$ in the end, I will now change the notation: Let the first decomposition be $f = \hat{u}_0 + v_0$. Next, this “too large” \hat{u}_0 is decomposed downwards to $\hat{u}_0 = \hat{u}_{-1} + u_0$ where \hat{u}_{-1} represents the smaller (but possibly still too large) BV part and the residual u_0 only contains the information between scale $1/2\lambda_0$ and $1/\lambda_0$. Continuing this decomposition iteratively, the entire reversed initialisation procedure is described by

$$\hat{u}_\ell = \hat{u}_{\ell-1} + u_\ell, \quad [\hat{u}_{\ell-1}, u_\ell] = \arg \min_{\hat{u}+u=\hat{u}_\ell} J(\hat{u}_\ell, 2^{\ell-1}\lambda_0) \quad (4.10)$$

for $\ell = 0, -1, \dots, -m_0$. For comparison: In the “forward procedure” $J(v_{\ell-1}, 2^\ell \lambda_0)$ was used, so the component at step ℓ is now evaluated with half the scaling factor. Also, now the texture containing residual (measured by $\|\cdot\|_2^2$) and not the BV part is kept, but since $\hat{u}_0 \in BV(\Omega)$, the “backward residuals” u_ℓ are still of bounded variation.

As ℓ decreases and the scale $\lambda_\ell = 2^\ell \lambda_0$ is continuously reduced, the procedure always decomposes the last (and thereby smallest) BV function \hat{u}_ℓ to an even “smoother” version $\hat{u}_{\ell-1}$ and the sought residual u_ℓ ; until after $m_0 \in \mathbb{N}$ steps, also the least oscillating part of f is exhausted by satisfying $\|f\|_* \leq 1/\lambda_{-m_0}$. Then, \hat{u}_{-m_0-1} and everything thereafter would equal zero, so the backward procedure is complete.

Now, the following multiscale decomposition is obtained:

$$\begin{aligned} f &= \hat{u}_0 + v_0 \\ &= \hat{u}_{-1} + u_0 + v_0 \\ &\vdots \\ &= u_{-m_0} + u_{-m_0+1} + \dots + u_0 + v_0, \end{aligned}$$

After this, the “forward procedure” (4.3) can be applied, starting with $v_0 = u_1 + v_1$ as usual. In the end, the multiscale representation $f \sim \sum_{\ell=-m_0}^m u_\ell$ with residual v_m is obtained.

A comparison of the concept of forward and backward procedure is sketched in Figure 4.1, as well as the distribution of information in the final hierarchical decomposition. There, the various elements of f are illustrated as oscillations of different frequencies. This corresponds to the interpretation of the *ROF*

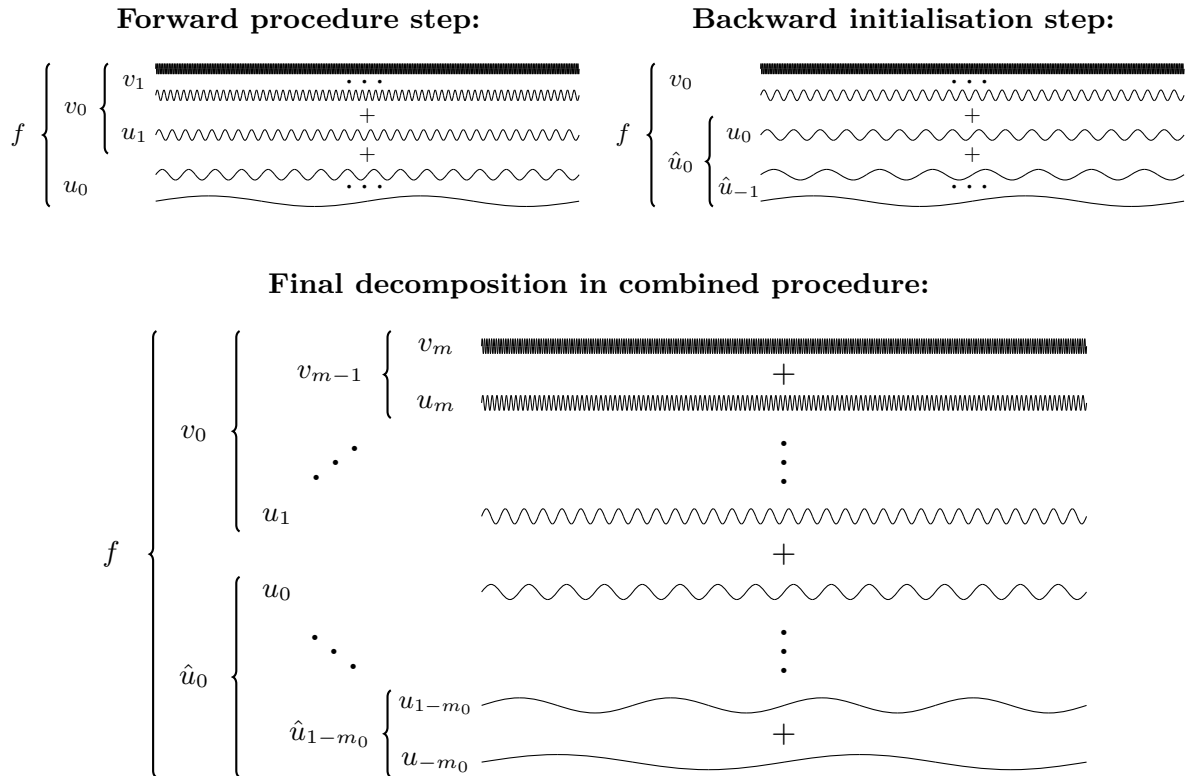


Figure 4.1: Top: Schematic visualisation of the image components contained in the decomposition after two steps in the forward vs. backward approach. Bottom: Distribution of components in the final hierarchical decomposition. The differently oscillating lines represent the different elements of an image: Noise and texture is associated with high oscillations, and the “most essential features” extracted at a low scale with rather little oscillation.

model as separation between highly oscillating components such as noise and texture, and the “essential features” characterised by small oscillations.

I finally want to note two aspects regarding this initialisation method: If the mean of f in Ω is not zero, then it would be contained in the last (or rather first) instance of u , namely u_{-m_0} . The initialising “backward procedure” should be ended once u_{-m_0} does not significantly consist of more information than the mean of f , that is $\|u_{-m_0}\|_{BV} \approx 0$. Furthermore, the results from Theorem 4.1, Theorem 4.2 and Lemma 4.3 are only valid for the “forward procedure”. If the *backward initialisation* is used, these results hold starting at $\ell = 0$ and with \hat{u}_0 instead of u_0 . Nevertheless, this has no impact on the convergence behaviour itself.

Visual results on the initialisation procedure are briefly discussed in Section 6.2, see especially Figure 6.9.

5

Discretisation and implementation of the models

In the previous chapters, I discussed mainly theoretical aspects of the *ROF model* and the multiscale representation obtained by hierarchically decomposing a signal f . But in this thesis, I also want to take a look at the application of both procedures to real images, and I will present some numerical results and analyses later in Chapter 6.

In the next part, I will first give some details on the notation used for the further discrete considerations and procedures, and describe the framework (particularly the grid) in detail (see Section 5.1). Afterwards, I will present the used algorithms for the one-time *ROF decomposition* (Section 5.2), and carry out their extension to the hierarchical case (Section 5.3).

5.1 Notation and Grids

In the following, I will only consider two-dimensional greyscale images of rectangular size, that is $f \in L^2(\Omega)$ where $\Omega = [a, b] \times [c, d] \subset \mathbb{R}^2$ – usually $\Omega = [0, 1]^2$. The image $f : \Omega \rightarrow [0, 1]$ is a real function on Ω whose values range from 0 (black) to 1 (white). Of course, in practice the image f is only known by a finite number of information in a discrete framework.

In this thesis, I use the discretisation model outlined in [17, Section 3.1]. More precisely, I assume that the data (information on f) is given on a grid of pixels. If N_1 and N_2 denote the number of pixels with respect to the first and second dimension, then the pixel length and thereby spatial distance between two neighbouring grid points is everywhere given by $h_1 = (b - a)/N_1$ and $h_2 = (d - c)/N_2$, respectively.

I consider the used grid to be *cell-centred*, meaning the intensity given for a pixel actually represents the one at the centre of that pixel. Thus, all grid points lie in those very pixel centres, so the x -coordinates of the grid points x_1, \dots, x_{N_1} are given by $x_i = a + (i - \frac{1}{2})h_1$ for $i = 1, \dots, N_1$, and simultaneously $y_j = c + (j - \frac{1}{2})h_2$ for $j = 1, \dots, N_2$. An illustration of the used grid is given in Figure 5.1.

In the following, let $f_{i,j} := f(x_i, y_j)$ and $u_{i,j} := u(x_i, y_j)$ denote the discrete information of f resp. u given at the corresponding grid points. Furthermore, from now on the denominators f and u will represent the original (and continuous) signals as well as their discrete models. Of course, mathematically both objects are different, but their identification should not lead to misunderstandings here and is useful in terms of readability and interpretation. I want to point out that even though in my programmes, the images are stored in 2D-arrays (like matrices), in a mathematical sense, the discrete signals f and u remain vectors of finite dimension $N_1 \cdot N_2$.

An approximation on the integral $\int_{\Omega} f(x)dx$ that is consistent with the *cell-centred grid*, is given by

$$\int_{\Omega} f(x)dx \approx h_1 h_2 \sum_{i,j} f_{i,j}, \quad \text{hence} \quad \|f\|_2^2 \approx h_1 h_2 \sum_{i,j} f_{i,j}^2 \quad \text{and} \quad \langle f, g \rangle \approx h_1 h_2 \sum_{i,j} f_{i,j} g_{i,j}.$$

Next, I will address the question of differentiation: Let the usual discrete *forward*, *backward* and *centred difference operators* be denoted by D_+ , D_- and D_0 , respectively. More precisely, the differences

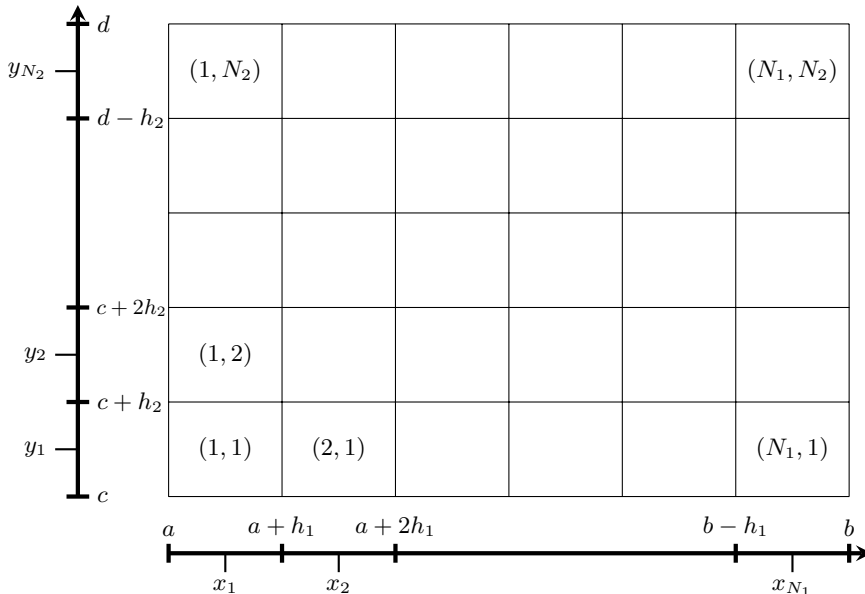


Figure 5.1: Illustration of the cell-centred grid used in this thesis for the domain $\Omega = [a, b] \times [c, d]$, [cf. 17, Figure 3.2]

in x - and y -direction are defined as [cf. 22]

$$\begin{aligned} D_+^x u_{i,j} &:= \frac{u_{i+1,j} - u_{i,j}}{h_1} & D_-^x u_{i,j} &:= \frac{u_{i,j} - u_{i-1,j}}{h_1} & D_0^x u_{i,j} &:= \frac{u_{i+1,j} - u_{i-1,j}}{2h_1} \\ D_+^y u_{i,j} &:= \frac{u_{i,j+1} - u_{i,j}}{h_2} & D_-^y u_{i,j} &:= \frac{u_{i,j} - u_{i,j-1}}{h_2} & D_0^y u_{i,j} &:= \frac{u_{i,j+1} - u_{i,j-1}}{2h_2}. \end{aligned}$$

In general, the central difference D_0 is often considered less suitable for approximating the actual first derivative because – even though it uses information from both sides – it misses thin structures and does not allow for oscillations of high frequencies [cf. 12, page 77] as it does not depend on $u_{i,j}$.

When it comes to approximating the total variation of a signal u , one can find several approaches for implementing this in the literature, see e.g. [12] and [8]. In this thesis, I will use the so-called *isotropic TV* with the *one-sided forward difference* D_+^x and D_+^y following [4, Section 3.1]:

$$\|u\|_{BV} \approx h_1 h_2 \sum_{i,j} \sqrt{(D_+^x u_{i,j})^2 + (D_+^y u_{i,j})^2}. \quad (5.1)$$

This is in accordance with Lemma 2.6, by which the total variation of a weakly differentiable function u is given by $\|u\|_{BV} = \int_{\Omega} |Du(x)| dx$. Furthermore, the TV discretisation (5.1) is consistent, meaning it converges to the continuous TV as the resolution becomes infinitely fine (i.e. $h_1, h_2 \rightarrow 0$) as stated in [4, Proposition 3.1]. Hence, I will approximate the energy functional $\mathcal{E}_{f,\lambda}$ by:

$$\mathcal{E}_{f,\lambda}(u) = \|u\|_{BV} + \lambda \|f - u\|_2^2 \approx h_1 h_2 \sum_{i,j} \sqrt{(D_+^x u_{i,j})^2 + (D_+^y u_{i,j})^2} + \lambda h_1 h_2 \sum_{i,j} (f_{i,j} - u_{i,j})^2. \quad (5.2)$$

5.2 Implementation of the ROF model

Quoting [6, page 14], numerous numerical algorithms have been proposed to approximate the minimiser u_{λ} of (ROF), most of them fall into the three main approaches “direct optimisation”, “solving the associated

Euler-Lagrange equation” and “using the dual variable”. Besides those, also many other algorithms were developed [cf. 12, page 78]. In this thesis, however, I will limit the discussion to the concept of solving the *Euler-Lagrange equation* associated with (ROF), as already the proposed algorithm of Rudin, Osher and Fatemi [20] and the one used by Tadmor, Nezzar and Vese [22] aimed for.

Discrete realisation of the Euler-Lagrange equation

The approximation to the solution u_λ of (ROF) is tackled by seeking a function $u \in BV(\Omega)$ that satisfies the associated *Euler-Lagrange equation* (3.11) (see Section 3.3). A numerical issue in this equation is faced whenever $Du = 0$, and especially since the occurrence of such “flat regions” is very likely [cf. 4, page 284], at first the singularity arising from the denominator in eq. (3.11) should be removed. Therefore, instead of minimising the functional $\mathcal{E}_{f,\lambda}(u) = \int_\Omega (|Du(x)| + \lambda(f(x) - u(x))^2) dx$ (see eq. (3.10)), the regularised energy functional

$$\mathcal{E}_{f,\lambda}^{(\varepsilon)}(u) := \int_\Omega \left(\sqrt{\varepsilon^2 + |Du(x)|^2} + \lambda(f(x) - u(x))^2 \right) dx \quad (5.3)$$

with $\varepsilon^2 > 0$ is being minimised [cf. 22, Section 4.2]. Clearly, $\mathcal{E}_{f,\lambda}^{(\varepsilon)}$ approaches $\mathcal{E}_{f,\lambda}$ as $\varepsilon \rightarrow 0$ and dissolves the singularity, but I should mention that in [6] it is stated that the use of an ε which is large enough for effective regularisation will reduce the ability of the *ROF model* to preserve edges [cf. 6, page 15]. Nevertheless, this concept will be used here and the same computations as in Section 3.3 now yield the regularised *Euler-Lagrange equation*

$$0 \stackrel{!}{=} u - f - \frac{1}{2\lambda} \operatorname{div} \left(\frac{Du}{\sqrt{\varepsilon^2 + |Du|^2}} \right). \quad (5.4)$$

Now, a discrete counterpart to this is needed, i.e. an equation which the discrete signals f and u must satisfy in conformance with (5.4). In Section 5.1, I introduced notations for discretely approximating the spatial differences D^x and D^y . Based on that, the following symmetrically balanced 2D discretisation of $\operatorname{div} \left(\frac{Du}{\sqrt{\varepsilon^2 + |Du|^2}} \right) = \frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{\varepsilon^2 + (u_x)^2 + (u_y)^2}} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{\varepsilon^2 + (u_x)^2 + (u_y)^2}} \right)$ is described in [22] (and similarly in [20] and [12]):

$$\begin{aligned} \frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{\varepsilon^2 + (u_x)^2 + (u_y)^2}} \right) \Big|_{(x_i, y_j)} &\approx D_-^x \left(\frac{D_+^x u_{i,j}}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_0^y u_{i,j})^2}} \right) \\ &= \frac{1}{h_1} D_-^x \left(\frac{u_{i+1,j} - u_{i,j}}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_0^y u_{i,j})^2}} \right) \\ &= \frac{1}{h_1^2} \left(\frac{u_{i+1,j} - u_{i,j}}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_0^y u_{i,j})^2}} - \frac{u_{i,j} - u_{i-1,j}}{\sqrt{\varepsilon^2 + (D_-^x u_{i,j})^2 + (D_0^y u_{i-1,j})^2}} \right), \end{aligned} \quad (5.5)$$

and simultaneously for y :

$$\begin{aligned} \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{\varepsilon^2 + (u_x)^2 + (u_y)^2}} \right) \Big|_{(x_i, y_j)} &\approx D_-^y \left(\frac{D_+^y u_{i,j}}{\sqrt{\varepsilon^2 + (D_0^x u_{i,j})^2 + (D_+^y u_{i,j})^2}} \right) \\ &= \frac{1}{h_2^2} \left(\frac{u_{i,j+1} - u_{i,j}}{\sqrt{\varepsilon^2 + (D_0^x u_{i,j})^2 + (D_+^y u_{i,j})^2}} - \frac{u_{i,j} - u_{i,j-1}}{\sqrt{\varepsilon^2 + (D_0^x u_{i,j-1})^2 + (D_-^y u_{i,j})^2}} \right). \end{aligned}$$

By introducing the following notation for the denominators

$$\begin{aligned} c_{i,j}^E &:= \frac{1}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_0^y u_{i,j})^2}} & c_{i,j}^W &:= \frac{1}{\sqrt{\varepsilon^2 + (D_-^x u_{i,j})^2 + (D_0^y u_{i-1,j})^2}} \\ c_{i,j}^N &:= \frac{1}{\sqrt{\varepsilon^2 + (D_0^x u_{i,j})^2 + (D_+^y u_{i,j})^2}} & c_{i,j}^S &:= \frac{1}{\sqrt{\varepsilon^2 + (D_0^x u_{i,j-1})^2 + (D_-^y u_{i,j})^2}}, \end{aligned} \quad (5.6)$$

a discrete approximation of eq. (5.4) in (x_i, y_j) is given by

$$0 \stackrel{!}{=} u_{i,j} - f_{i,j} - \frac{1}{2\lambda h_1^2} \left(c_{i,j}^E (u_{i+1,j} - u_{i,j}) - c_{i,j}^W (u_{i,j} - u_{i-1,j}) \right) - \frac{1}{2\lambda h_2^2} \left(c_{i,j}^N (u_{i,j+1} - u_{i,j}) - c_{i,j}^S (u_{i,j} - u_{i,j-1}) \right).$$

Rearranging this equation, multiplying by $2\lambda h_1^2 h_2^2$ and solving for $u_{i,j}$, finally yields:

$$\begin{aligned} 0 &\stackrel{!}{=} u_{i,j} \left(1 + \frac{c_{i,j}^E + c_{i,j}^W}{2\lambda h_1^2} + \frac{c_{i,j}^N + c_{i,j}^S}{2\lambda h_2^2} \right) - f_{i,j} - \frac{c_{i,j}^E u_{i+1,j} + c_{i,j}^W u_{i-1,j}}{2\lambda h_1^2} - \frac{c_{i,j}^N u_{i,j+1} + c_{i,j}^S u_{i,j-1}}{2\lambda h_2^2} \\ \Leftrightarrow u_{i,j} &\stackrel{!}{=} \frac{2\lambda h_1^2 h_2^2 f_{i,j} + h_2^2 (c_{i,j}^E u_{i+1,j} + c_{i,j}^W u_{i-1,j}) + h_1^2 (c_{i,j}^N u_{i,j+1} + c_{i,j}^S u_{i,j-1})}{2\lambda h_1^2 h_2^2 + h_2^2 (c_{i,j}^E + c_{i,j}^W) + h_1^2 (c_{i,j}^N + c_{i,j}^S)}. \end{aligned} \quad (5.7)$$

Note that the superscript of the constants stands for the corresponding cardinal direction and indicates their position in the grid relative to (x_i, y_j) . This, together with a depiction of the information used for the computation of each constant, is illustrated in Figure 5.2.

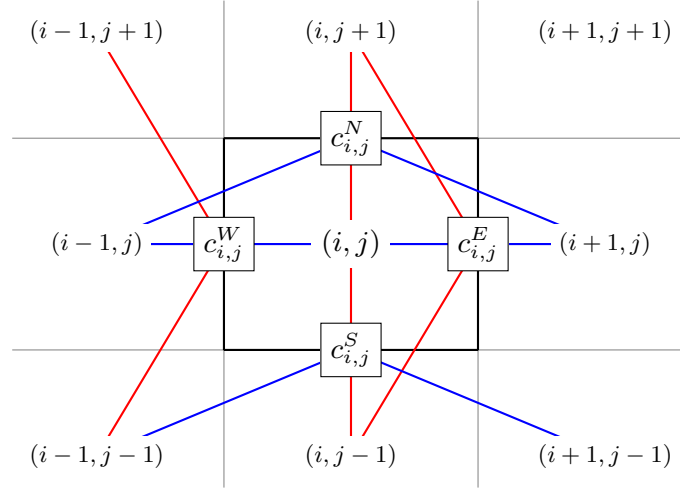


Figure 5.2: Schematic location of constants $c_{i,j}^E$, $c_{i,j}^W$, $c_{i,j}^N$ and $c_{i,j}^S$ in the grid relative to the bold pixel (x_i, y_j) . The blue and red lines illustrate which neighbouring grid points are used for the computation of each constant in x and y direction, respectively.

As stated in (3.12), the *Euler-Lagrange equation* is augmented by *Neumann boundary conditions*, so that there is no flux at the boundary going in- or outwards the system. Numerically, this is met by employing *reflection boundary treatment* to u [22, page 566], meaning: On each side of the grid, an additional border line is added to that grid containing the same values as the neighbouring line, as illustrated in Figure 5.3. Specifically, the grid $\{x_1, \dots, x_{N_1}\} \times \{y_1, \dots, y_{N_2}\}$ is extended to $\{x_0, \dots, x_{N_1+1}\} \times \{y_0, \dots, y_{N_2+1}\}$ where for $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$:

$$\begin{aligned} u_{0,N_2+1} &= u_{1,N_2} & u_{i,N_2+1} &= u_{i,N_2} & u_{N_1+1,N_2+2} &= u_{N_1,N_2} \\ u_{0,j} &= u_{1,j} & & & u_{N_1+1,j} &= u_{N_1,j} \\ u_{0,0} &= u_{1,1} & u_{i,0} &= u_{i,1} & u_{N_1+1,0} &= u_{N_1,1} \end{aligned} \quad (5.8)$$

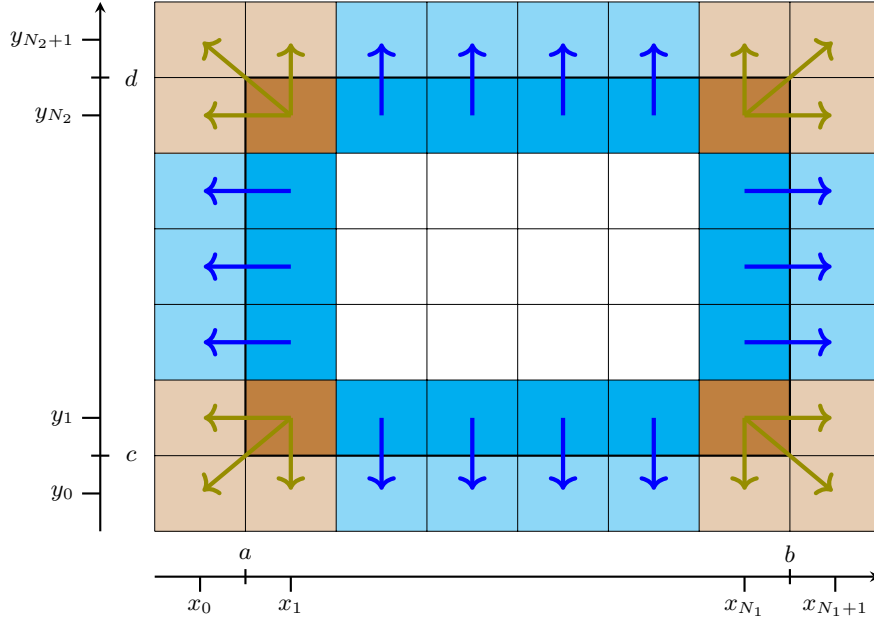


Figure 5.3: Illustration of the reflection boundary treatment in the grid corresponding to Figure 5.1.

Alternative derivation of the discrete Euler-Lagrange equation

The above approach to derive eq. (5.7) by discretising the continuous *Euler-Lagrange equation* (5.4) has a major disadvantage: The latter requires that u_λ is weakly differentiable and even the divergence of its (weak) gradient properly defined (cf. Section 3.3). But in fact, it is possible to arrive at eq. (5.7) without this requirement or any limitations to the continuous setting, as I will show next.

The idea is as follows: Instead of first deriving the *Euler-Lagrange equation* associated with the continuous functional $\mathcal{E}_{f,\lambda}^{(\varepsilon)}$ and afterwards discretising this equation, the functional itself is discretised directly, so its minimisation then corresponds to a finite dimensional optimisation process. In the coming considerations, I will assume the functional $\mathcal{E}_{f,\lambda}^{(\varepsilon)}$ of the regularised *ROF problem* to be discretised analogously to eq. (5.2), that is

$$\mathcal{E}_{f,\lambda}^{(\varepsilon)}(u) \approx h_1 h_2 \sum_{i,j} \sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_+^y u_{i,j})^2} + \lambda h_1 h_2 \sum_{i,j} (f_{i,j} - u_{i,j})^2. \quad (5.9)$$

To illustrate the following manipulations more clearly in “matrix-vector form”, the vector $u = (u_{i,j})$ will be reshaped to $\mathbf{u} = \left(\underbrace{u_{1,1}, \dots, u_{N_1,1}}_{u_{i,1}}, \underbrace{u_{1,2}, \dots, u_{N_1,2}}_{u_{i,2}}, \dots, \underbrace{u_{1,N_2}, \dots, u_{N_1,N_2}}_{u_{i,N_2}} \right)^T \in \mathbb{R}^{N_1 N_2}$ and the same way f to $\mathbf{f} \in \mathbb{R}^{N_1 N_2}$. Thus, grid points neighbouring in x -dimension are denoted directly below each other, whereas neighbours in y -dimension are associated with distance N_1 . At first, I note that now $\sum_{i,j} (f_{i,j} - u_{i,j})^2 = (\mathbf{f} - \mathbf{u})^T I (\mathbf{f} - \mathbf{u})$.

From now on, the first non-zero row and column will always denote the element corresponding to the (i, j) -component. To describe $D_+^x u_{i,j}$ as a matrix-vector product in terms of \mathbf{u} , I define the operator

$$D_{i,j}^x := \frac{1}{h_1} (0, \dots, 0, \underbrace{-1}_{(i,j)^{\text{th}}}, 1, 0, \dots, 0), \quad \text{so that} \quad D_{i,j}^x \mathbf{u} = D_+^x u_{i,j} = \frac{u_{i+1,j} - u_{i,j}}{h_1}.$$

Thus, it follows:

$$\mathbf{u}^T (D_{i,j}^x)^T D_{i,j}^x \mathbf{u} = \mathbf{u}^T \underbrace{\frac{1}{h_1^2} \begin{pmatrix} 0 & & & & \\ & \ddots & & & \\ & & 1 & -1 & \\ & & -1 & 1 & 0 \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix}}_{=: D_{i,j}^{2x}} \mathbf{u} = \frac{1}{h_1^2} \mathbf{u}^T \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -u_{i+1,j} + u_{i,j} \\ u_{i+1,j} - u_{i,j} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \frac{u_{i+1,j}^2 - 2u_{i+1,j}u_{i,j} + u_{i,j}^2}{h_1^2}$$

and hence $(D_+^x u_{i,j})^2 = (D_{i,j}^x \mathbf{u})^2 = \mathbf{u}^T D_{i,j}^{2x} \mathbf{u}$.

Similarly, $D_{i,j}^y$ is defined where $D_{i,j}^y \mathbf{u} = D_+^y u_{i,j} = \frac{u_{i,j+1} - u_{i,j}}{h_2}$ and $(D_+^y u_{i,j})^2 = (D_{i,j}^y \mathbf{u})^2 = \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}$. With this notation, (5.9) takes the form

$$\mathcal{E}_{\mathbf{f},\lambda}^{(\varepsilon)}(\mathbf{u}) \approx h_1 h_2 \sum_{i,j} \sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}} + \lambda h_1 h_2 (\mathbf{f} - \mathbf{u})^T I (\mathbf{f} - \mathbf{u}). \quad (5.10)$$

Since $\mathcal{E}_{\mathbf{f},\lambda}^{(\varepsilon)}$ is continuously differentiable in \mathbf{u} , the necessary condition for optimality (see e.g [18, Theorem 2.2]), that is for \mathbf{u} to possibly be a minimiser, is given by $\nabla \mathcal{E}_{\mathbf{f},\lambda}^{(\varepsilon)}(\mathbf{u}) \stackrel{!}{=} 0$. Thus, the differentiation of each component of (5.10) with respect to \mathbf{u} is sought:

Therefore, at first it is $\nabla(\mathbf{f} - \mathbf{u})^T I (\mathbf{f} - \mathbf{u}) = 2(\mathbf{u} - \mathbf{f})$, and since $D_{i,j}^{2x}$ and $D_{i,j}^{2y}$ are both symmetric, also $\nabla \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} = 2D_{i,j}^{2x} \mathbf{u}$ resp. $\nabla \mathbf{u}^T D_{i,j}^{2y} \mathbf{u} = 2D_{i,j}^{2y} \mathbf{u}$. Hence, for one entire “square root term” it holds:

$$\nabla \sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}} = \frac{D_{i,j}^{2x} \mathbf{u} + D_{i,j}^{2y} \mathbf{u}}{\sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}}}.$$

But this yields (for simplicity I will only discuss the x -dimension):

$$\begin{aligned} \sum_{i,j} \frac{D_{i,j}^{2x} \mathbf{u}}{\sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}}} &= \sum_{i,j} \frac{1}{h_1^2} \frac{1}{\sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}}} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -u_{i+1,j} + u_{i,j} \\ u_{i+1,j} - u_{i,j} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\ &= \frac{1}{h_1^2} \begin{pmatrix} \vdots \\ \frac{u_{i,j} - u_{i-1,j}}{\sqrt{\varepsilon^2 + \mathbf{u}^T D_{i-1,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i-1,j}^{2y} \mathbf{u}}} - \frac{u_{i+1,j} - u_{i,j}}{\sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}}} \\ \vdots \end{pmatrix} \\ &= -\frac{1}{h_1} \begin{pmatrix} \vdots \\ D_-^x \frac{u_{i+1,j} - u_{i,j}}{\sqrt{\varepsilon^2 + \mathbf{u}^T D_{i,j}^{2x} \mathbf{u} + \mathbf{u}^T D_{i,j}^{2y} \mathbf{u}}} \\ \vdots \end{pmatrix} \\ &= -\begin{pmatrix} \vdots \\ D_-^x \frac{D_+^x u_{i,j}}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_+^y u_{i,j})^2}} \\ \vdots \end{pmatrix} \end{aligned}$$

Except for D_+^y instead of D_0^y – this last expression completely equals (5.5) which was the discretisation of the “ x -dimensional divergence part” of the continuous *Euler-Lagrange equation*.

Altogether, one finally obtains that the necessary condition $\nabla \mathcal{E}_{f,\lambda}^{(\varepsilon)}(\mathbf{u}) \stackrel{!}{=} 0$ is equivalent to:

$$0 \stackrel{!}{=} u_{i,j} - f_{i,j} - \frac{1}{2\lambda} \left(D_-^x \frac{D_+^x u_{i,j}}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_+^y u_{i,j})^2}} + D_-^y \frac{D_+^y u_{i,j}}{\sqrt{\varepsilon^2 + (D_+^x u_{i,j})^2 + (D_+^y u_{i,j})^2}} \right) \forall i, j,$$

and with the same manipulations as above, one arrives precisely at eq. (5.7) – with the only difference that the central discrete differences D_0^x and D_0^y are replaced by the forward ones D_+^x and D_+^y , respectively.

In view of Figure 5.2, this means the constants $c_{i,j}^E$ and $c_{i,j}^N$ coincide into a “Northeastern one”, $c_{i,j}^W$ moves to the “Northwest” and $c_{i,j}^S$ to the “Southeast”, so the scheme loses the symmetry of eq. (5.7). Hence, in the further considerations and my code, I continue with the latter.

In conclusion, the results of this section show that the “*Euler-Lagrange*-based” approach that lead to eq. (5.7), is actually also a proper strategy to deal with (ROF) when solely imposing that the functional to be minimised is discretely approximated by eq. (5.9). Assumptions on the differentiability of u up to second order as required for the continuous *Euler-Lagrange equation* in (5.4), however, are not necessary.

An iterative method to approximate the ROF minimiser

In this thesis, I use a *Gauss-Seidel fixed point iterative method* to approximate a solution of eq. (5.4), as also done in [22]. In general, this *fixed point procedure* is known to properly work and converge to a solution only for linear systems [21]. Since eq. (5.4) is non-linear in the first place because of its denominator, it is “linearised by lagging the diffusion coefficient” [6, page 14]: In every step, $\sqrt{\varepsilon^2 + |Du|^2}$ is a priori estimated – based on the previous information. Specifically, in the iteration over $n \in \mathbb{N}_0$, the $(n+1)$ -th iterate is obtained by solving the linear equation

$$0 = u^{(n+1)} - f - \frac{1}{2\lambda} \operatorname{div} \left(\frac{Du^{(n+1)}}{\sqrt{\varepsilon^2 + |Du^{(n)}|^2}} \right).$$

In the discrete setting of eq. (5.7), this “lagging of the diffusion coefficient” corresponds to computing the constants $c_{i,j}^E, c_{i,j}^W, c_{i,j}^N, c_{i,j}^S$ before the $(n+1)$ -th iteration based on the information of step n , namely $u_{i,j}^{(n)}, u_{i-1,j}^{(n)}$ and $u_{i,j-1}^{(n)}$ as well as their neighbouring points for the discrete differences. Then, eq. (5.7) is a linear system since (despite its notation as 2D-array) u is a vector of size $N_1 \cdot N_2$, and $f_{i,j}$ and the $c_{i,j}$ ’s are given numbers for $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$.

Gauss-Seidel methods generally aim at solving a quadratic system of linear equations, say $Au - b = 0$ where u is the unknown. When decomposing the linear operator $A = D + L + U$ into its diagonal components D , the lower triangular part L and the upper triangular part U , it should hold $(D + L)u = b - Uu$. Furthermore should be assumed that D is invertible, i.e. no diagonal component of A is zero. Now, the idea of *Gauss-Seidel iterative* methods is to approximate a solution of $Au - b = 0$ by iteratively updating u as result of the equivalent equation $u = (D + L)^{-1}(b - Uu)$. That is, when initialising with $u^{(0)}$ and iterating over $n \in \mathbb{N}_0$, in particular: $u^{(n+1)} = (D + L)^{-1}(b - Uu^{(n)})$. For more on iterative methods, see e.g. [21, Chapter 4].

This concept can now directly be projected to solving the linear system of eq. (5.7). There, the denominator represents the (positive, thus non-zero) diagonal components of A , b is given by $2\lambda h_1^2 h_2^2 f$ and the rest corresponds to diagonals of a sparse linear operator, i.e. to components of L and U . Hence, by setting $u_{i,j}$ and the already computed elements $u_{i-1,j}$ and $u_{i,j-1}$ to $u_{i,j}^{(n+1)}, u_{i-1,j}^{(n+1)}$ resp. $u_{i,j-1}^{(n+1)}$, and fixing $u_{i+1,j}$ and $u_{i,j+1}$ at time step n , the iterative procedure given by

$$u_{i,j}^{(n+1)} = \frac{2\lambda h_1^2 h_2^2 f_{i,j} + h_2^2 (c_{i,j}^E u_{i+1,j}^{(n)} + c_{i,j}^W u_{i-1,j}^{(n+1)}) + h_1^2 (c_{i,j}^N u_{i,j+1}^{(n)} + c_{i,j}^S u_{i,j-1}^{(n+1)})}{2\lambda h_1^2 h_2^2 + h_2^2 (c_{i,j}^E + c_{i,j}^W) + h_1^2 (c_{i,j}^N + c_{i,j}^S)} \quad (5.11)$$

is a *Gauss-Seidel method* for approximately solving eq. (5.7). Since this procedure uses the most recent information on u , it requires to compute each $u^{(n)}$ element by element, iterating over $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$.

In conformance with the reflection boundary condition, after every step n the additional border lines are filled with the values from the neighbouring grid points as described in (5.8). The procedure ends either after a given maximum number of iterations n_{max} , or as in [22], when no significant update is obtained any more, meaning that for a preassigned tolerance τ_{stop} it holds:

$$\left\| u^{(n)} - u^{(n-1)} \right\|_2^2 \leq \tau_{stop}. \quad (5.12)$$

Instead of considering the actual change in $u^{(n)}$, one could also measure the latest improvement in terms of the objective function $\mathcal{E}_{f,\lambda}$, so a second possible stopping criterion would be

$$\mathcal{E}_{f,\lambda} \left(u^{(n-1)} \right) - \mathcal{E}_{f,\lambda} \left(u^{(n)} \right) \leq \tau_{stop} \quad (5.13)$$

An alternative approach arises from Theorem 3.4: According to that, the minimising pair $[u_\lambda, v_\lambda]$ satisfies $\langle u_\lambda, v_\lambda \rangle = \|u_\lambda\|_{BV} / 2\lambda$. So to test if a resultant $u^{(n)}$ is rather close to the solution u_λ , it can be checked whether this equation is approximately met. Hence, once $|\langle u_\lambda, v_\lambda \rangle - \|u_\lambda\|_{BV} / 2\lambda| \leq \tau_{stop}$, the iteration will be terminated. In the discrete setting, this computation can be tackled by:

$$\begin{aligned} \left| \langle u, f - u \rangle - \frac{1}{2\lambda} \|u\|_{BV} \right| &\approx \left| h_1 h_2 \sum_{i,j} u_{i,j} (f_{i,j} - u_{i,j}) - \frac{1}{2\lambda} h_1 h_2 \sum_{i,j} \sqrt{(D_+^x u)_{i,j}^2 + (D_+^y u)_{i,j}^2} \right| \\ &= \left| h_1 h_2 \sum_{i,j} \left(u_{i,j} (f_{i,j} - u_{i,j}) - \frac{1}{2\lambda} \sqrt{(D_+^x u)_{i,j}^2 + (D_+^y u)_{i,j}^2} \right) \right| \stackrel{?}{\leq} \tau_{stop} \end{aligned} \quad (5.14)$$

In Section 6.1, I will present some observations made with these three quantities, see especially Figure 6.3. In either case, let n_{stop} denote the final iteration step at which the stopping criterion is met, so the iteration runs for $n = 0, \dots, n_{stop}$. Then $u^{(n_{stop})}$ represents an accurate approximation of (5.4)'s “fixed point steady solution” u_λ on the grid, i.e. $u_{i,j}^{(n_{stop})} \approx u_\lambda(x_i, y_j)$ [cf. 22, page 567].

Finally, I briefly want to address the question how to initialise the procedure, i.e. how $u^{(0)}$ is chosen. Lemma 3.2 yields that for the minimiser u_λ it must hold $\int_\Omega u_\lambda = \int_\Omega f$. Transferred to a discrete scenario and due to the *reflection boundary treatment*, $u^{(0)}$ should satisfy $h_1 h_2 \sum_{i,j} u_{i,j}^{(0)} = h_1 h_2 \sum_{i,j} f_{i,j}$.

The usual strategy that was e.g. also used in [22], is to set $u_{i,j}^{(0)} := f_{i,j}$. This leads to a resultant $u^{(n_{stop})}$ that is normally rather close to the original signal (especially in case of slow convergence). However, it seems desirable to me to choose the starting values in a way that the initial decomposition $[u^{(0)}, f - u^{(0)}]$ is a priori at least not worse than the trivial decomposition $[0, f]$, meaning $\mathcal{E}_{f,\lambda}(u^{(0)}) = \|u^{(0)}\|_{BV} + \lambda \|f - u^{(0)}\|_2^2 \not\leq \lambda \|f\|_2^2 = \mathcal{E}_{f,\lambda}(0)$. But for rather “small” λ this may very well happen when choosing $u^{(0)} := f$. This brought me to the idea of using the “optimal combination” of f and $\bar{f} := \frac{1}{N_1 N_2} \sum_{i,j} f_{i,j}$, meaning: $\alpha \in [0, 1]$ is chosen such that the objective function $\mathcal{E}_{f,\lambda}(u^{(0)})$ is minimal for $u^{(0)} = \alpha f + (1 - \alpha)\bar{f}$. Analytically:

$$\mathcal{E}_{f,\lambda}(\alpha f + (1 - \alpha)\bar{f}) = \|\alpha f + (1 - \alpha)\bar{f}\|_{BV} + \lambda \|f - \alpha f - (1 - \alpha)\bar{f}\|_2^2 = \alpha \|f\|_{BV} + \lambda(1 - \alpha)^2 \|f - \bar{f}\|_2^2$$

which is a convex parabola attaining its minimum when

$$0 = \|f\|_{BV} - 2\lambda(1 - \alpha) \|f - \bar{f}\|_2^2 \quad \Leftrightarrow \quad \alpha = 1 - \frac{\|f\|_{BV}}{2\lambda \|f - \bar{f}\|_2^2}.$$

Thus, in addition to the choice $u^{(0)} := f$, I propose using $u^{(0)} := \alpha f + (1 - \alpha)\bar{f}$ where

$$\alpha := \max \left\{ 1 - \frac{\|f\|_{BV}}{2\lambda \|f - \bar{f}\|_2^2}, 0 \right\} \in [0, 1]. \quad (5.15)$$

Algorithm 1 Gauss-Seidel iterative method to approximate the (ROF) minimiser

Given: $(f_{i,j})$ where $i = 1, \dots, N_1$, $j = 1, \dots, N_2$ and $\lambda > 0$
Choose: $(u_{i,j}^{(0)})$, $\varepsilon^2 > 0$, $\tau_{stop} > 0$ and $n_{max} \in \mathbb{N}$
 \triangleright e.g. $u_{i,j}^{(0)} \leftarrow f_{i,j} \quad \forall i, j$
 $n_{stop} \leftarrow n_{max}$

 Built additional border lines in $u^{(0)}$ for reflection boundary treatment as described in (5.8)

for $n \leftarrow 0$ to n_{max} **do**
for $i \leftarrow 1$ to N_1 **do**
for $j \leftarrow 1$ to N_2 **do**

 Compute constants $c_{i,j}^E, c_{i,j}^W, c_{i,j}^N, c_{i,j}^S$ based on $u^{(n)}$ as described in (5.6)

 Compute $u_{i,j}^{(n+1)}$ as described in (5.11)

end for
end for

 Update additional border line in $u^{(n+1)}$ as described in (5.8)

if $\|u^{(n+1)} - u^{(n)}\|_2^2 \leq \tau_{stop}$ **then**
 \triangleright Alternatively use (5.13) or (5.14)

 $n_{stop} \leftarrow n + 1$
break
end if
end for
Return: $u^{(n_{stop})}$

In case of small values for α , one may end up with a rather blurred, grey appearing result. The value of α also gives an idea on the nature of the parameter λ or in particular how much of f is interpreted as *texture* at that scale, as α already gives information about whether u is “better” if it is very smooth or rather similar to f . In practice, however, the choice $u^{(0)} := f$ appeared to not only yield visually better results but also a faster minimisation of $\mathcal{E}_{f,\lambda}$. A reason for this could be that large areas with only few variation lead to very small updates since the constants in (5.11) become dominant then. I will briefly discuss results on both methods in Section 6.1, see especially Figure 6.2.

A precise description of the entire procedure that I used for the numerical experiments in this thesis is given in Algorithm 1. But what about its properties of this *fixed point iterative method* for the linearised *Euler-Lagrange equation*? The first follows from (5.11): There, every summand in the numerator corresponds to one in the denominator, with the difference that it comes with always exactly one component of f , $u^{(n)}$ or $u^{(n+1)}$. Hence, it holds everywhere and for $n \in \mathbb{N}$ that

$$u_{i,j}^{(n)} \leq \max_{i,j} \{f_{i,j}, u_{i,j}^{(0)}\} \quad \text{and} \quad u_{i,j}^{(n)} \geq \min_{i,j} \{f_{i,j}, u_{i,j}^{(0)}\}.$$

With the above discussed choices for $u^{(0)}$, this yields that if f is bounded by the constants m_0 and M_0 in the sense $m_0 \leq f_{i,j} \leq M_0$ for all i, j , then also the estimate u is bounded by these constants in every iteration n , i.e. $m_0 \leq u_{i,j}^{(n)} \leq M_0$. Hence, the procedures lead at most to a “shrinking” or “smoothing” of the original signal. This property is also in agreement with the maximum principle for the continuous case, by which for the minimiser u_λ of (ROF) it holds $m_0 \leq u_\lambda(x) \leq M_0$ if $m_0 \leq f(x) \leq M_0$ for all $x \in \Omega$ [cf. 16, page 36]. Furthermore, in [6, page 15] it is mentioned that these fixed point iterative methods converge to the actual (ROF) minimiser, even though only with linear rate; but empirically already after few iterations a result of “visual accuracy” is obtained. A more precise statement on their convergence was first developed by Chan and Mulet in 1999 [7].

At last, I consider it worth noting that in the expectable case of quadratic pixels, that is $h_1 = h_2 =: h$, the above computations simplify quite a bit: Then, eq. (5.11) can be reduced to the form [cf. 22, page 566]

$$u_{i,j}^{(n+1)} = \frac{2\lambda h^2 f_{i,j} + c_{i,j}^E u_{i+1,j}^{(n)} + c_{i,j}^W u_{i-1,j}^{(n+1)} + c_{i,j}^N u_{i,j+1}^{(n)} + c_{i,j}^S u_{i,j-1}^{(n+1)}}{2\lambda h^2 + c_{i,j}^E + c_{i,j}^W + c_{i,j}^N + c_{i,j}^S}.$$

Furthermore, when replacing ε^2 by $\varepsilon^2 h^2$ in the computation of the constants, then the division by h in every discrete difference can be omitted and instead the entire fraction multiplied by h ; this may speed up the numerics noticeably.

5.3 Implementation of the hierarchical decomposition

After having a procedure to approximate the optimal *ROF decomposition* $[u_\lambda, v_\lambda] = \arg \min_{u+v=f} J(f, \lambda)$ for a given signal $f \in BV(\Omega)$ and fixed parameter $\lambda > 0$, this is now used for computing the *hierarchical* (BV, L^2) *decomposition* of Chapter 4 which results in a *multiscale image representation* $f = \sum_{\ell=0}^m u_\ell + v_m$, or when using the “backward initialisation” of Section 4.3, in $f = \sum_{\ell=-m_0}^m u_\ell + v_m$.

Starting with the computation of a first, coarse decomposition $f = u_0 + v_0$ at scale $\lambda_0 > 0$ following Algorithm 1 where $u_0 := u^{(n_{stop})}$, the *hierarchical decomposition* of the “forward procedure” $[u_\ell, v_\ell] = \arg \min_{u+v=v_{\ell-1}} J(v_{\ell-1}, 2^\ell \lambda_0)$ is obtained by reiterating this process for $\ell = 1, \dots, m$ in the manner [22]

$$\begin{cases} f_{new} & \leftarrow & f_{old} - u^{(n_{stop})} \\ \lambda_{new} & \leftarrow & 2\lambda_{old}. \end{cases}$$

In every decomposition step ℓ , a new initial guess $u_\ell^{(0)}$ needs to be chosen. For this, the above strategies *input data* and *optimal combination* can be used – in my numerical experiments, initialising with the input data gave the best results.

On the question when the entire multiscale procedure should be ended, namely the determination of m , are three options presented in [22]: The first is to directly pre-specify the number of scales m . In view of Theorem 4.1, this corresponds to measuring the amount of ultimately remaining texture since $\|v_m\|_* = 1/2\lambda_m$. An alternative is to measure the energy $\|u_m\|_2^2$ below a certain tolerance to ensure that a significant update is obtained in every step (Note: In the original paper $\|u_m - u_{m-1}\|_2^2$ is given [cf. 22, page 567], but in my opinion that criterion makes little sense and should supposedly describe only the update). Their last proposal regards the dissolving of texture: Once $\|v_{m-1}\|_2^2 - \|v_m\|_2^2$ becomes sufficiently small, the iteration is ended. The entire “forward procedure” to obtain the multiscale representation of f for given m and λ_0 with scaling factor 2 is described in Algorithm 2.

Algorithm 2 Forward procedure to compute a hierarchical (BV, L^2) decomposition

Given: $(f_{i,j})$ where $i = 1, \dots, N_1$, $j = 1, \dots, N_2$, $m \in \mathbb{N}$ and $\lambda_0 > 0$

Choose: Initialisation and stopping strategy for each ROF decomposition

Approximate $[u_0, v_0] = \arg \min_{u+v=f} J(f, \lambda_0)$ using Algorithm 1 where: $u_0 \leftarrow u^{(n_{stop})}$ and $v_0 \leftarrow f - u_0$

for $\ell \leftarrow 1$ to m **do**

Approximate $[u_\ell, v_\ell] = \arg \min_{u+v=v_{\ell-1}} J(v_{\ell-1}, 2^\ell \lambda_0)$ using Algorithm 1 where $u_\ell \leftarrow u^{(n_{stop})}$ and

$$v_\ell \leftarrow v_{\ell-1} - u_\ell$$

▷ Optional: Check stopping criteria

end for

Return: (u_0, \dots, u_m)

The last question that needs to be addressed is the initialisation of the hierarchical decomposition. To distribute the information of f to several different instances, clearly $\lambda_0 > 0$ should be chosen “rather small”. In Section 4.3, however, I introduced an approach to initialise the hierarchical (BV, L^2) decomposition such that the information of possibly missing scales are captured in a “backward initialisation procedure”: In this, the first BV result \hat{u}_0 is iteratively decomposed by $[\hat{u}_{\ell-1}, u_\ell] = \arg \min_{\hat{u}+u=\hat{u}_\ell} J(\hat{u}_\ell, 2^{\ell-1} \lambda_0)$

until $\|\hat{u}_{-m_0}\|_{BV} \approx 0$. Afterwards, the “normal forward decomposition” is continued. A description of this enhanced procedure is given in Algorithm 3. It is consistent with the schematic illustration of the hierarchical decomposition sketched in Figure 4.1. Visual results on this procedure are briefly discussed in Section 6.2, see especially Figure 6.9.

Algorithm 3 Forward procedure with backward initialisation to compute a hierarchical (BV, L^2) decomposition

Given: $(f_{i,j})$ where $i = 1, \dots, N_1$, $j = 1, \dots, N_2$, $m, m_0 \in \mathbb{N}$ and $\lambda_0 > 0$

Choose: $\tau_{init} > 0$, initialisation and stopping strategy for each ROF decomposition

Approximate $[\hat{u}_0, v_0] = \arg \min_{\hat{u}+v=f} J(f, \lambda_0)$ using Algorithm 1 where $\hat{u}_0 \leftarrow u^{(n_{stop})}$ and $v_0 \leftarrow f - \hat{u}_0$

for $\ell \leftarrow 0$ to $-m_0 + 1$ **do**

Approximate $[\hat{u}_{\ell-1}, u_\ell] = \arg \min_{\hat{u}+u=\hat{u}_\ell} J(\hat{u}_\ell, 2^{\ell-1}\lambda_0)$ using Algorithm 1 where $\hat{u}_{\ell-1} \leftarrow u^{(n_{stop})}$ and

$$u_\ell \leftarrow \hat{u}_\ell - \hat{u}_{\ell-1}$$

if $\|\hat{u}_{\ell-1}\|_{BV} \leq \tau_{init}$ **then**

$$m_0 \leftarrow -\ell + 1$$

break

end if

end for

$$u_{-m_0} = \hat{u}_{-m_0}$$

▷ Last residual becomes u_{-m_0}

for $\ell \leftarrow 1$ to m **do**

Approximate $[u_\ell, v_\ell] = \arg \min_{u+v=v_{\ell-1}} J(v_{\ell-1}, 2^\ell \lambda_0)$ using Algorithm 1 where $u_\ell \leftarrow u^{(n_{stop})}$ and

$$v_\ell \leftarrow v_{\ell-1} - u_\ell$$

▷ Optional: Check stopping criteria

end for

Return: $(u_{-m_0}, \dots, u_0, \dots, u_m)$

6

Numerical results

In the last chapter of this thesis, I present some results from the numerical experiments I made: First in Section 6.1, I will discuss some observations regarding the one-time *ROF decomposition* of an image as described in Chapter 3. This includes the behaviour and iterative development of the *fixed point iterative method* from Section 5.2, the influence of the choice of initial data $u^{(0)}$ and weighting parameter λ , and the decomposition of some exemplary images. Afterwards in Section 6.2, I will discuss results on the hierarchical (BV, L^2) decomposition leading to a *multiscale representation* of images as described in Chapter 4. There, I focus on the resolving of texture throughout the decomposition steps, and present a visual analysis of the *backward initialisation* of Section 4.3.

For my numerical experiments, I implemented the methods described in Chapter 5 for approximating the solution of (ROF) and computing a hierarchical decomposition. All code is written and executed in MATLAB (Version R2020b) [15] and published on GitHub [13].

6.1 Results on the one-time ROF decomposition

If not mentioned otherwise, for the different computations I used the *Gauss-Seidel scheme* described in Section 5.2, with initialisation $u_{i,j}^{(0)} := f_{i,j}$ and stopping tolerance $\tau_{stop} = 10^{-8}$ for stopping strategy “update size” (5.12). All computations are done with regularising parameter $\varepsilon^2 = 10^{-6}$.

Figure 6.1 shows the process of the *ffjord picture’s ROF decomposition* with the results $u^{(n)}$ and $v^{(n)}$ obtained after $n = 1, 15, 50$ and 300 steps for two different values of λ . Furthermore, the development of the target functional $\mathcal{E}_{f,\lambda}$ and its components $\|u\|_2^2$ and $\lambda \|v\|_2^2$ during these 300 iteration steps is plotted. In order to have the programme run for 300 iterations, τ_{stop} was set to 10^{-11} here.

It can be seen that between all four iteration instances, some changes occur and the method does not “arrive” precisely at the (ROF) minimiser after few iterations. However, despite some more information being transferred to v , the visual difference between the results after 50 and 300 steps appears rather small to me. Thus, the choice of a smaller tolerance $\tau_{stop} = 10^{-8}$ by which the procedures end after 63 resp. 30 iterations seems reasonable – also in view of the obtained progress in the target functional $\mathcal{E}_{f,\lambda}$.

When comparing the final results for the different values of λ , the image f was much stronger smoothed in case of the smaller λ than when decomposed at a larger scale. This meets the expectations of Chapter 3 and especially Theorem 3.4. Furthermore, the result in the “stronger smoothing” case $\lambda = 1 \cdot 10^3$ obtained after 15 steps looks quite similar to the final one for $\lambda = 4 \cdot 10^3$. This is numerically interesting when considering to compute only very few steps and instead choosing a smaller value for λ – at least when initialising with $u_{i,j}^{(0)} := f_{i,j}$.

Next, I want to give an example in which the functional $\mathcal{E}_{f,\lambda}$ evaluated at the initial data $u^{(0)}$ is smaller and thus the latter a priori a more “precise” starting value when choosing $u_{i,j}^{(0)} := \alpha f_{i,j}^{(0)} + (1 - \alpha)\bar{f}$ with “optimal parameter” $\alpha \in [0, 1]$ as in (5.15), instead of $u_{i,j}^{(0)} := f_{i,j}^{(0)}$ (i.e. $\alpha = 1$). This is the case for instance for the *ffjord picture* at $\lambda = 2 \cdot 10^2$ where the “optimal parameter” is given by $\alpha \approx 0.47$. The initialisation data and the approximated *ROF decomposition* after 50 steps is shown in Figure 6.2, as well as the development of $\mathcal{E}_{f,\lambda}$ and its components.

6 Numerical results

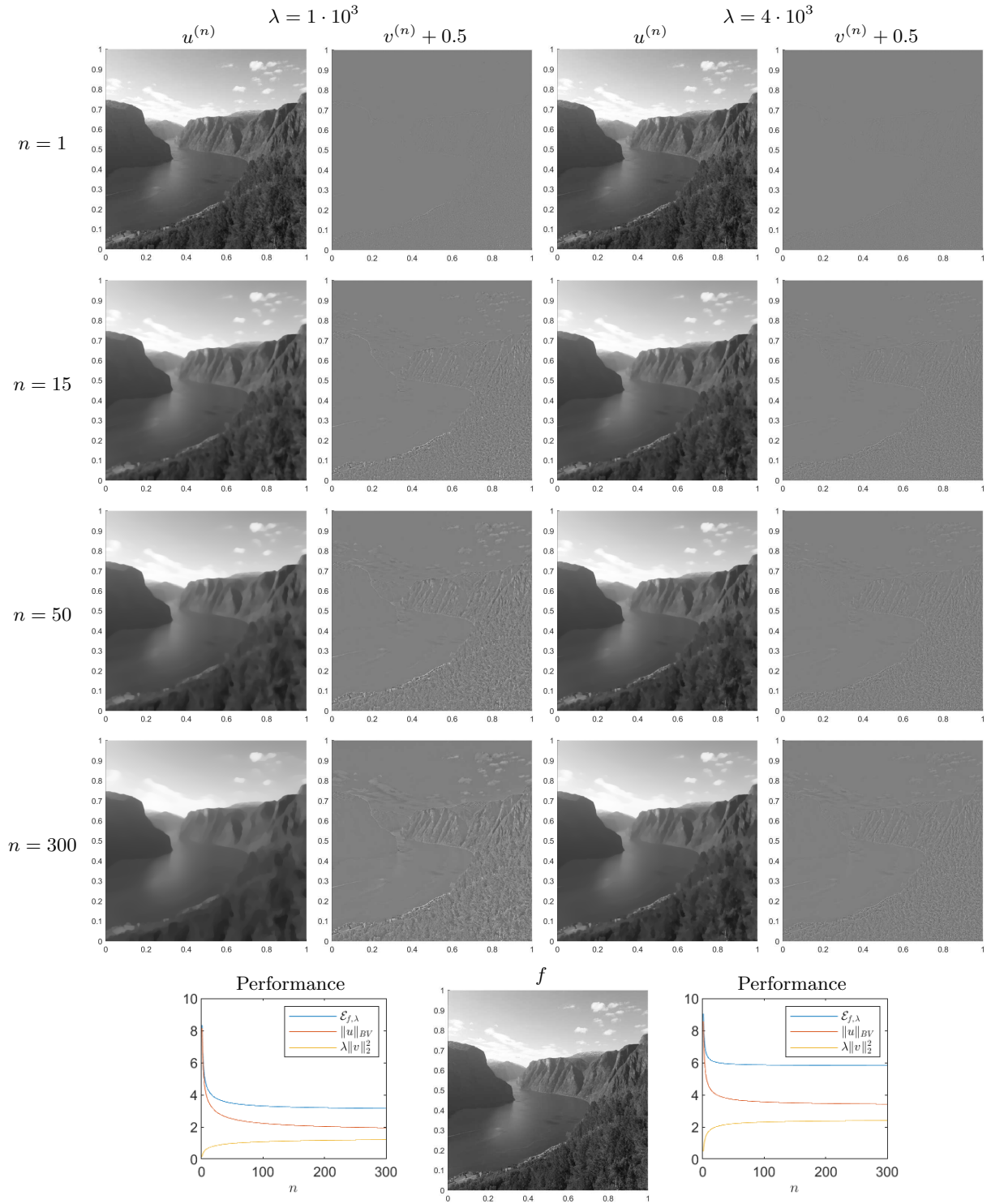


Figure 6.1: Approximation to the u_λ and v_λ components of the *ROF decomposition* obtained in the fixed point iterative procedure after $n = 1, 15, 50, 300$ iterations for parameter $\lambda = 1e3$ and $4e3$. At the bottom, the development of the target functional $\mathcal{E}_{f,\lambda}$ and its components during the iteration is shown.

6 Numerical results

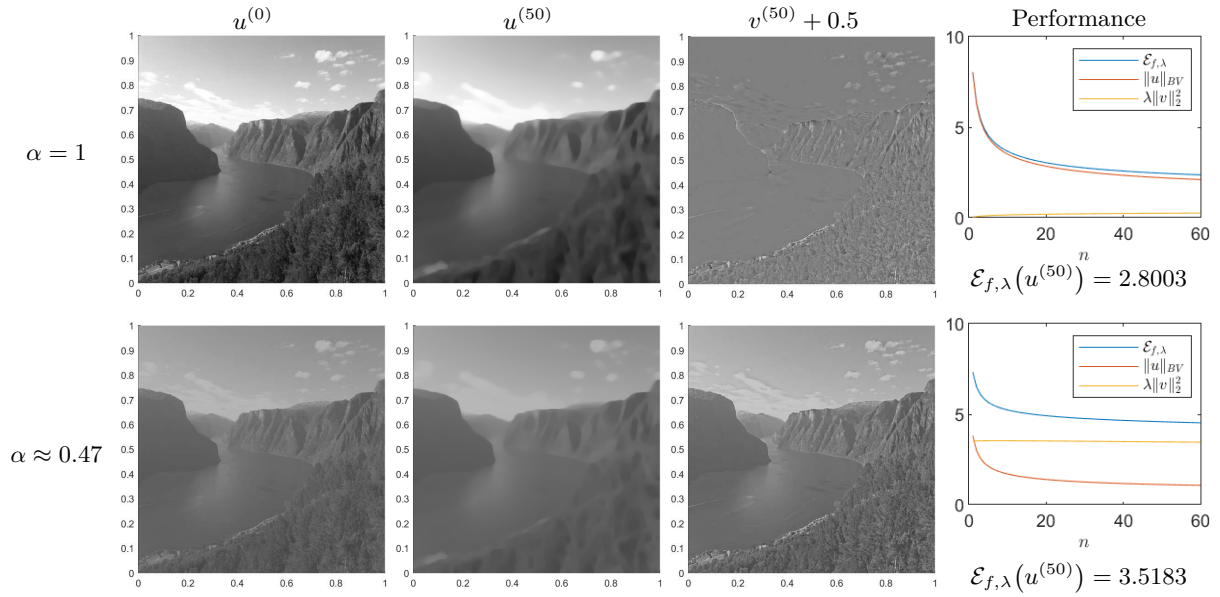


Figure 6.2: Approximation to the *ROF decomposition* for $\lambda = 2e2$ obtained after 50 steps when initialising with $u^{(0)} := f$ (top) vs. the “optimal combination” suggested in Section 5.2 (bottom). To the right, the development of the target functional $\mathcal{E}_{f,\lambda}$ and its components during the iteration is shown.

As the graphs show, in the beginning of the iteration, $\mathcal{E}_{f,\lambda}$ is smaller for the “optimal combination”. This is due to the smaller total variation in $u^{(0)}$ and despite the residual $v^{(0)}$ being relatively large. However, after a couple of iterations this turns as the total variation of u decreases rapidly in both cases, whereas the size of the residual remains almost unchanged for $\alpha \approx 0.47$. Thus – at least in this example – the fixed point iterative method performs well in smoothening the original image, but fails in recovering parts of it from a “too smooth” initialisation data. Instead, the “greyish” character of the initialisation data with $\alpha \approx 0.47$ remains even after 50 iterations almost unchanged.

So far, the behaviour of the used *fixed point iterative method* was only discussed in terms of the energy functional $\mathcal{E}_{f,\lambda}$. However, in Section 5.2 I mentioned that the accuracy of the computed approximation could also be viewed under Theorem 3.4 by measuring (5.14). The development of the latter together with the quantities “update size” (5.12) and “improvement in $\mathcal{E}_{f,\lambda}$ ” (5.13) in the decomposition of the *fjord picture* at $\lambda = 10^3$ is visualised in Figure 6.3, with logarithmic y-axes. Again, $\tau_{stop} = 10^{-11}$ here.

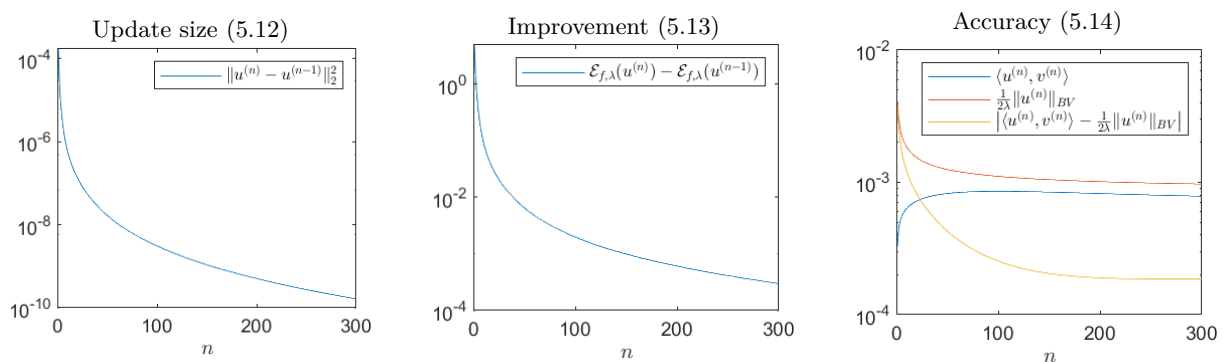


Figure 6.3: Development of the quantities “update size”, “improvement in $\mathcal{E}_{f,\lambda}$ ” and “accuracy in terms of Theorem 3.4” during the iteration. Under these, the convergence and performance of the procedure can be evaluated. The computation is from the decomposition of the *fjord picture* at scale $\lambda = 1e3$.

The graphs show, how – at least in this computation – the used *Gauss-Seidel scheme* does yield a more and more accurate result with increasing number of iterations. In my opinion, especially the last criterion gives a good impression on the behaviour of the procedure and that after at most 100 steps the optimal result is almost obtained, as it directly corresponds to the question, how well the (ROF) minimiser’s property of Theorem 3.4 is satisfied by its computed approximation.

Finally, I want to give a short notion on the applicability of the *ROF model* in general. In Figure 6.4, I present the computed (ROF) decompositions of different example images at manually chosen scales $\lambda > 0$: Since the minimisation of total variation is expected to remove highly oscillating components of a signal (cf. Example 2.9), the *ROF model* is especially suitable for denoising images – the field it also originates from [cf. 20]. Therefore, on the one hand, I show the decomposition of four noisy images. The resultant $u^{(n_{stop})}$ s represent denoised versions of them, whereas the residual $v^{(n_{stop})}$ mainly consists of the removed noise. The first two images are modifications of the *fjord picture*, to which I artificially added two different levels of *Gaussian noise*, the third corresponds to an example from [5] and mainly serves as object of comparison, and the last one is the noisy version of a square. It shows, how especially (almost) piecewise constant images are reconstructed well by the *ROF model*.

On the other hand, I want to focus on the interpretation of the *ROF model* as separation of “essential features” from texture. Therefore four images without artificial noise are decomposed at a relatively small scale. The first one again is the *fjord picture*, the second corresponds to an example from [22] which will be used more in Section 6.2, and the third is an x-ray image from the FAIR toolbox [17]. The last example shows the decomposition of a simple square without noise. Here, there is no texture to be extracted, instead it illustrates how the *ROF model* “moves” the edges of the square in a way that the shape approaches a circle. This behaviour of “object shifting and minimisation” is not surprising since the perimeter corresponds to the total variation $\|u\|_{BV}$ (see e.g. [11, Theorem 5.9]), and – in case of the square – is also obtained when analytically solving (ROF), as described for instance in [4, pages 292-293].

In my interpretation, the observations from Figure 6.4 show that the *ROF model* can give (at least visually) very useful results in the contexts of image reconstruction and denoising, as well as for the smoothening, simplification and texture extraction of images. But in some scenarios, it may also shift the edges of objects, change their form or reduce their size.

6.2 Results on the hierarchical (BV, L^2) image decomposition

Last but not least, I will present some results that are obtained when iteratively decomposing an image by (ROF) as described in Chapter 4 and Section 5.3. The resultant multiscale representation is exemplarily shown for three images: In Figure 6.5, the hierarchical decomposition of the *fjord picture* is given. It illustrates, how more and more information are captured with each refinement step: Starting with the coarse silhouettes of the scenery, until at last even the texture details of the trees in the bottom right corner are resolved. The remaining residual after 10 steps, v_9 , on the other hand is nearly empty, so in the end, the original image f is mainly reobtained.

In comparison to that, Figure 6.6 shows the hierarchical decomposition of a noisy version of the *fjord picture*. The noise – associated with oscillations of smallest period [22] – is kept longest in v and only starts reappearing in u_ℓ at higher scales when most texture is already contained in some previous instance of u . The first results, on the other hand, are relatively similar to the non-noisy version since the *ROF model* is effective especially in the reduction of noise. At $\ell = 5$, the residual v_ℓ finally consists mainly of noise, and after further iterations, the noisy image would be entirely recovered in u .

The third image is the main example from [22]. Its use is advantageous here: On the one hand to easier compare the results, but also because it consists of much differently fine texture which is resolved clearly visible step by step. In Figure 6.7, the different components of the image’s hierarchical (BV, L^2) decomposition at different levels of m are given with a simultaneous visualisation of the cumulative results $\sum_{\ell=0}^m u_\ell$, the sole new information u_m and the remaining residual v_m .

6 Numerical results

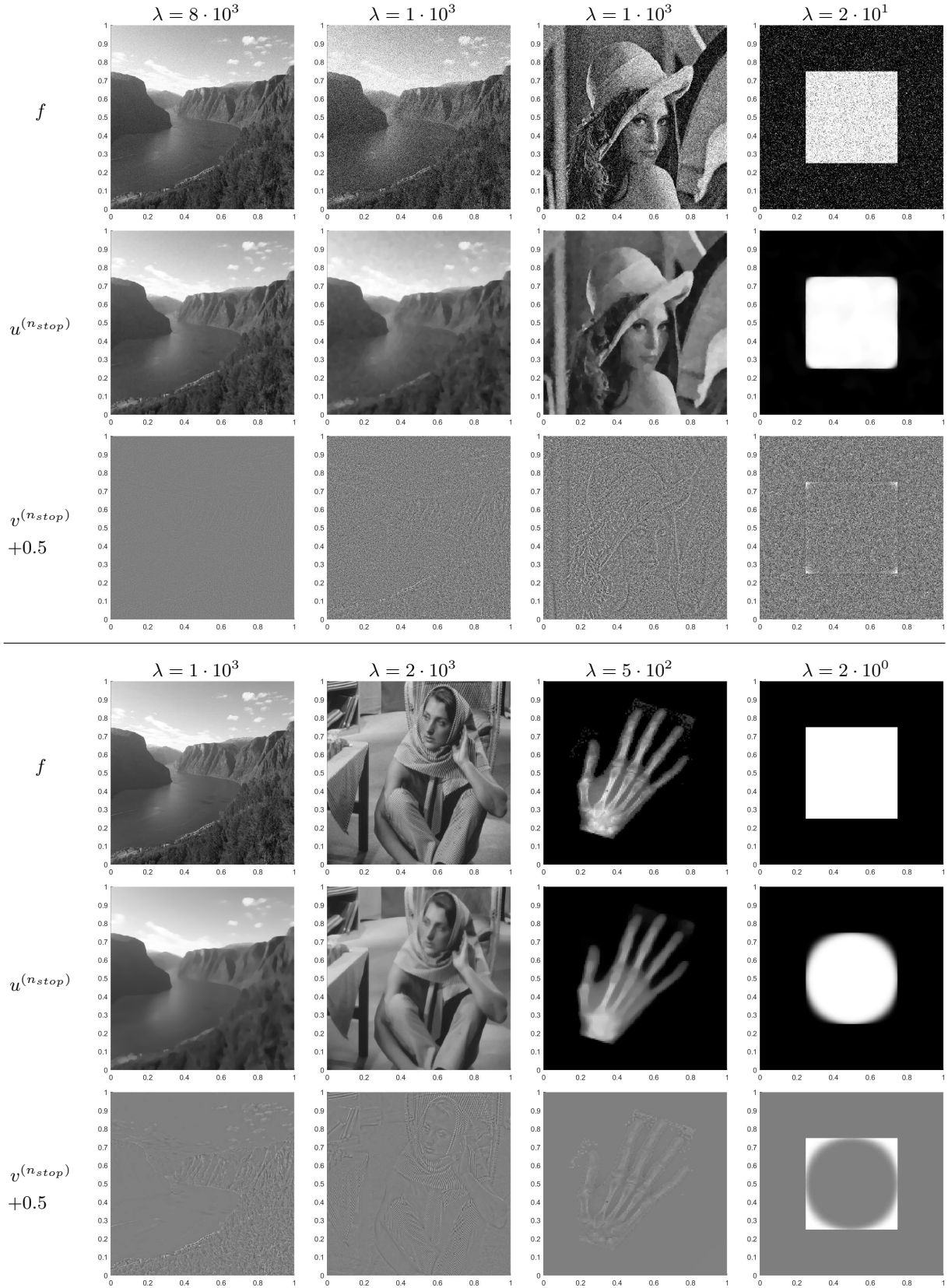


Figure 6.4: Approximated ROF decomposition of different images. The top four examples show the reconstruction of noisy images, and the four at the bottom the extraction of texture from “clean” images.

6 Numerical results

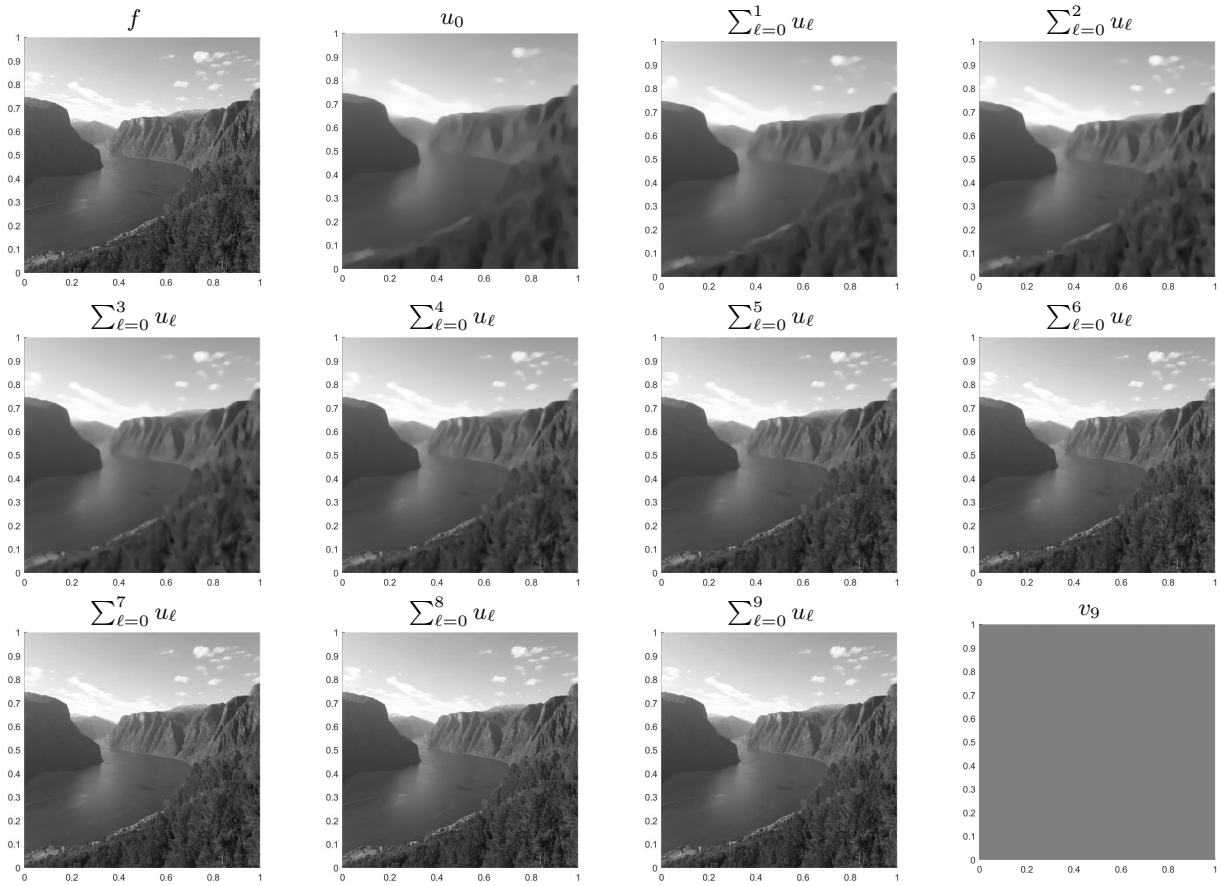


Figure 6.5: Hierarchical decomposition of the *fjord picture* for 10 steps with $\lambda_0 = 2e2$. In each step, more information are resolved and included in the resultant representation.

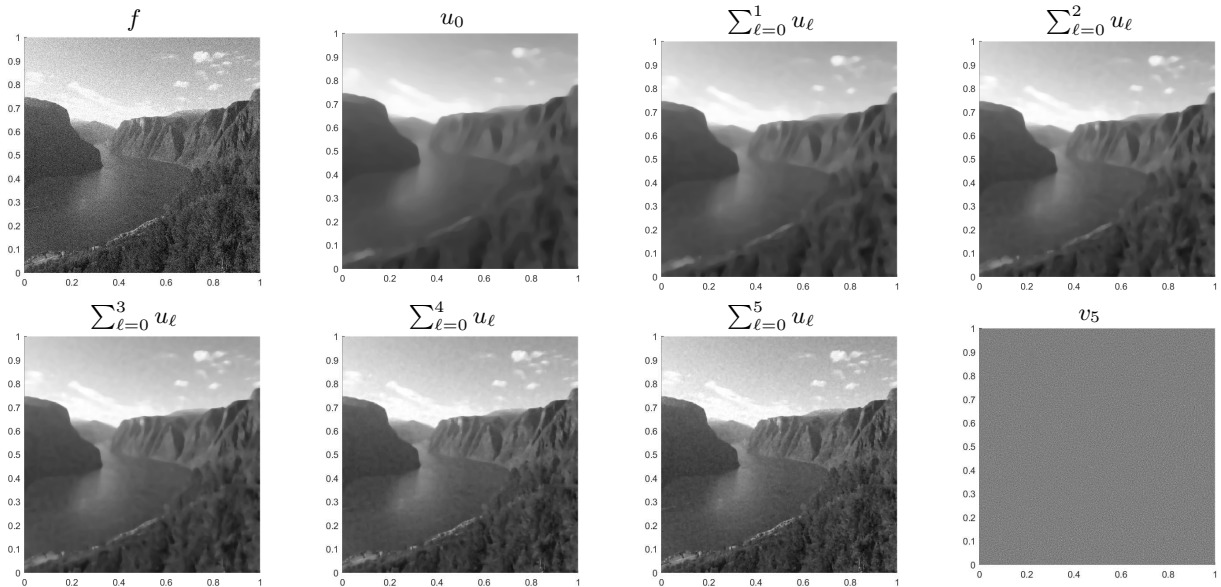


Figure 6.6: Hierarchical decomposition of a noisy version of the *fjord picture* for 6 steps with $\lambda_0 = 2e2$. At first, the noise is removed effectively and results resemble the ones of Figure 6.5, but after 4 steps when most texture was already resolved, the noise starts reappearing.

6 Numerical results

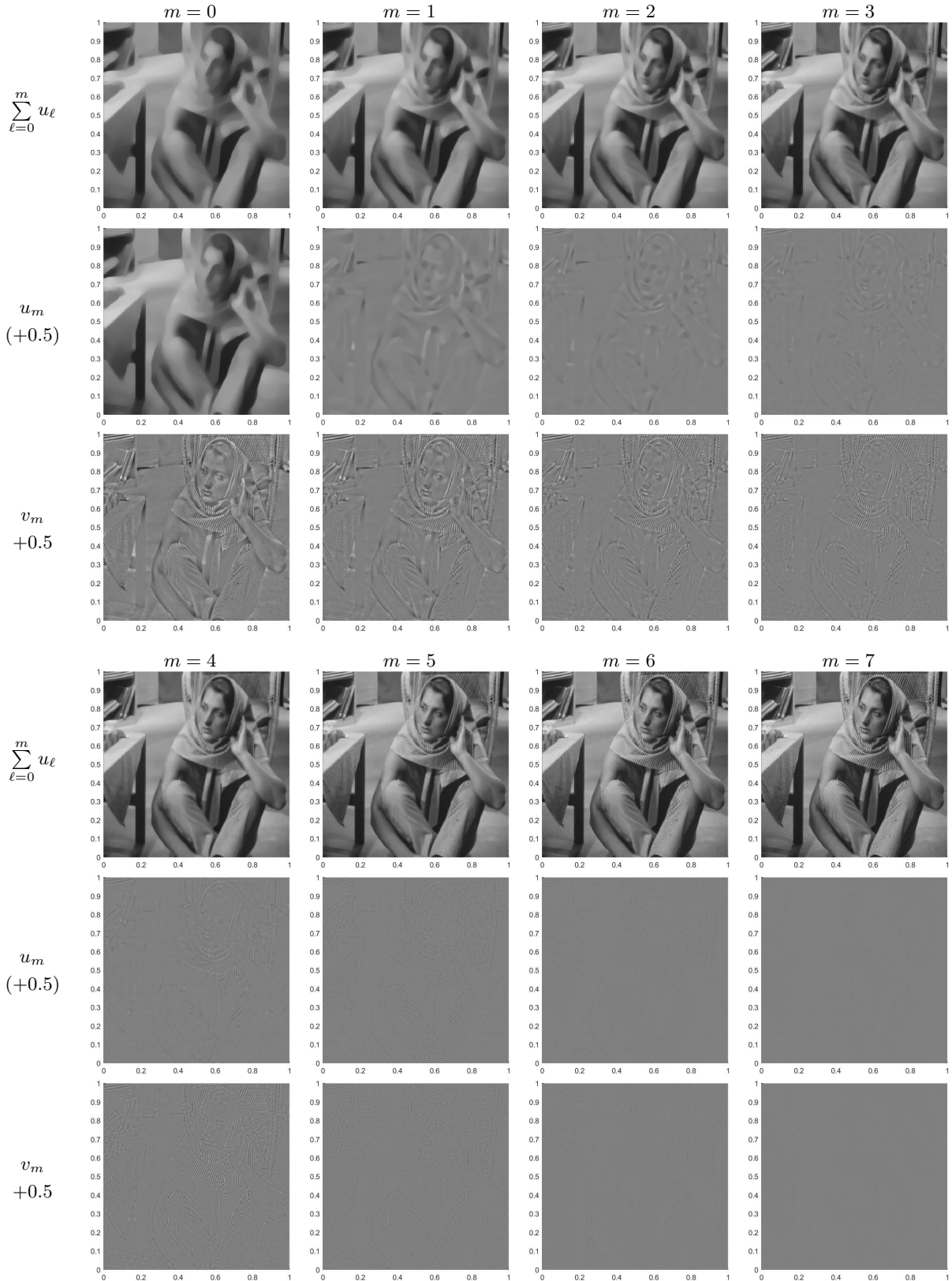


Figure 6.7: The different components (cumulative result, latest information, remaining residual) of the hierarchical decomposition of the picture of a woman, $\lambda_0 = 2e2$.

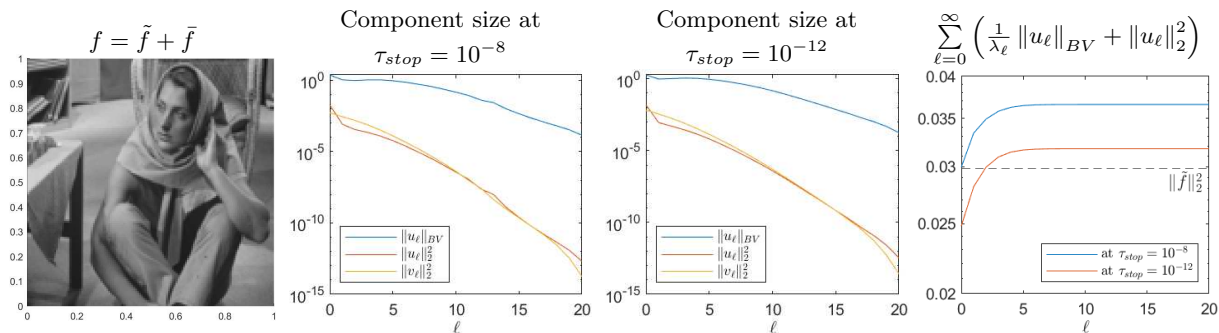


Figure 6.8: Middle: Development of total variation and energy of u_{ℓ} and v_{ℓ} during the iteration of the hierarchical decomposition of \tilde{f} for different stopping tolerances, $\lambda_0 = 2e2$. Right: Accuracy of the resultant decomposition for both tolerances in terms of the energy distribution of Theorem 4.2.

It illustrates how the various “texture elements” of any v_m are resolved bit by bit with every following newly scaled iteration, so the residual shrinks with every step, whereas the cumulative result gets closer and closer to the original image f . After eight iterations (at $m = 7$), the remaining residual is not noticeable any more. Note especially the development of texture detail at the top of the image where the different scales of the scarf’s and chair’s texture determine, at which step they are included in u .

In Section 4.2, I stated some properties on the convergence of $\sum_{\ell=0}^{\infty} u_{\ell}$ to f and the distribution of energy in the decomposition, especially that for $f \in BV(\Omega)$ it holds $\sum_{\ell=0}^{\infty} \left(\frac{1}{\lambda_{\ell}} \|u_{\ell}\|_{BV} + \|u_{\ell}\|_2^2 \right) = \|f\|_2^2$ (4.9). It is expectable that in case of the numerical approximations on each minimiser u_{ℓ} , this equation will not exactly hold – but the more accurate the approximations are (i.e. the smaller τ_{stop}), the smaller the discrepancy will be. Precisely this behaviour could be observed in my experiments, and is exemplarily illustrated in Figure 6.8 for the 128x128 pixel mean value adjusted image of the woman \tilde{f} . The mean value adjustment by which \tilde{f} takes values in $[-\bar{f}, 1 - \bar{f}]$, is done to reduce the size of $\|f\|_2^2$ and $\|u_0\|_2^2$, so the actual differences become more visible.

In fact, the less accurate the approximation, the more does the left hand side of eq. (4.9) exceed $\|\tilde{f}\|_2^2$. The reason for this supposedly is the initialisation $u_{i,j}^{(0)} := f_{i,j}^{(0)}$ in every step, yielding $u^{(n_{stop})}$ s with rather too high variation. Thus, “too many information” for the current scale λ_{ℓ} are contained in each u_{ℓ} . The quantities shown in Figure 6.8 also make up the possible criteria to end the iteration mentioned in Section 5.3: It was proposed to either measure $\|u_{\ell}\|_2^2$ or $\|v_{\ell} - 1\|_2^2 - \|v_{\ell}\|_2^2$ below some preassigned value. Visually, in Figure 6.7, v_{ℓ} appears “empty” after 10 steps, so afterwards no significant changes should occur, and this corresponds to the rapid decrease of $\|u_{\ell}\|_{BV}$, $\|u_{\ell}\|_2^2$ and $\|v_{\ell}\|_2^2$ after for $\ell \geq 10$.

Finally, I compare the plain forward decomposition of Algorithm 2 used so far with the results yielded by the additional backward initialisation described in Section 4.3 and Algorithm 3. To do so, I decompose the same image as before on the one hand five steps forward, starting at $\lambda_0 = 2 \cdot 10^2$ and finishing at $\lambda_4 = 3.2 \cdot 10^3$, and on the other hand three times backward and once forward, starting at $\lambda_0 = 1.6 \cdot 10^3$, thus scaling down until $\lambda_{-3} = 2 \cdot 10^2$ and afterwards going up to $\lambda_1 = 3.2 \cdot 10^3$. By that, the image is evaluated at the same scales in both cases, and should optimally result in very similar hierarchical decompositions. The results are visualised in Figure 6.9 and show, how well my proposed initialisation procedure for capturing possibly missing scales in this case works, as both procedures yield visually very similar images at each corresponding scale.

6 Numerical results

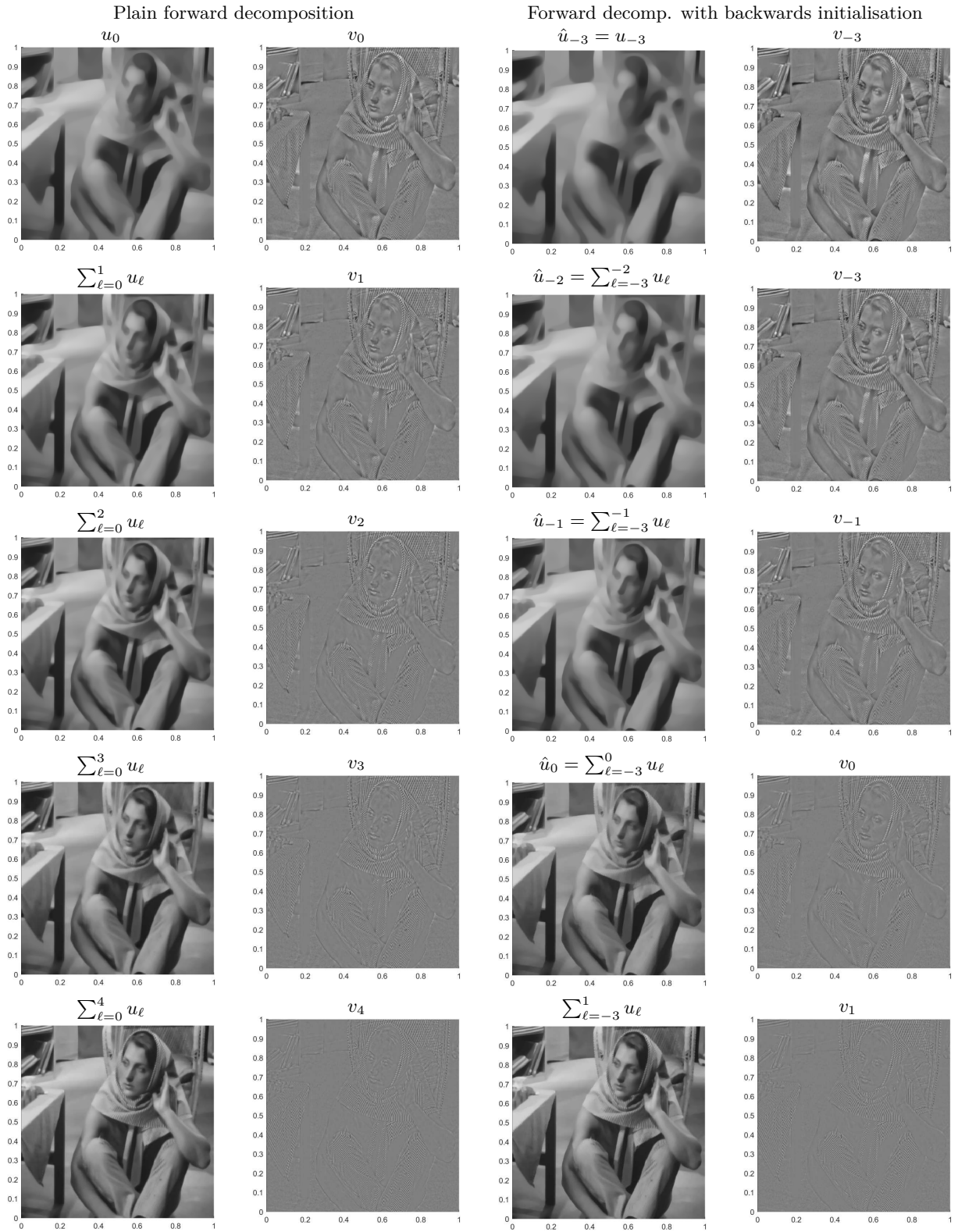


Figure 6.9: Cumulative results and residuals of the hierarchical decomposition at the same scales, when applying the procedure without (left) and with (right) backward initialisation. Here, $\lambda_0 = 2e2$ resp. $\lambda_0 = 1.6e3$.

7

Conclusion

In this thesis, I discussed why $BV(\Omega)$ is a proper space to represent “essential features” of an image $f \in L^2(\Omega)$, and how these may be extracted in an image decomposition $f = u_\lambda + v_\lambda$ [20]. The discussions in Chapters 2 and 3 and the results from Section 6.1 show that this *ROF model* on the one hand is effective in removing noise and other spurious oscillations from images, since these are associated with a high total variation as stated in [22] and described in Example 2.9. On the other hand, the minimisation of total variation yields a relatively simplified and smoothed image in terms of Lemma 2.12 b) that can be approximated by smooth functions (Lemma 2.12 a)). The values of u_λ remain in the range of f [16, page 36], and the *ROF model* does usually not “produce new discontinuities” [4, page 295].

However, most importantly in the context of this thesis is the following interpretation of the *ROF model*: Roughly speaking, by minimising (ROF), the information of an image that can properly be described and represented in the BV space – particularly edges and large homogeneous regions – are being separated from the rest, for instance from noise and texture. To this notion, Theorem 3.4 [cf. 16] gives an essential statement on the nature of the decomposition: It shows how the tuning parameter $\lambda > 0$ sets a cut in the *ROF model*’s distinction between “cartoonish parts” and “texture”. Image components “down to scale” $1/2\lambda$ are captured in u_λ , whereas the texture “below this scale” remains in the residual v_λ . Thus, the parameter λ determines which elements belong to which part of the decomposition, and Theorem 3.4 even quantifies this behaviour in terms of the dual norm $\|\cdot\|_*$.

In the second part of the thesis, the hierarchical (BV, L^2) decomposition of an image [22] was discussed. It yields a multiscale image representation $f \sim \sum_{\ell=0}^m u_\ell$ in which the various components of f are captured in different instances: Starting with a very simplified, “cartoonish” version of the image, its texture is resolved step by step, until the original image is finally reconstructed. In which instance u_ℓ each component ends, is thereby determined by the corresponding scale $\lambda_\ell = 2^\ell \lambda_0$: According to the results of Theorem 3.4, u_ℓ consists precisely of the elements of f which lie between scale $1/\lambda_\ell$ and $1/2\lambda_\ell$. The visual results from my experiments in Section 6.2 show, how this theoretical character of the hierarchical decomposition resolving finer texture in each step, can also be observed in practice.

Quoting [22], the sum $f \sim \sum_{\ell=0}^m u_\ell$ “provides a multilayered description of f ” which lies precisely in the “intermediate scale of spaces, in between BV and L^2 ”, and this multiscale (BV, L^2) expansion “is particularly suitable for image representations”. Furthermore, as discussed in Theorem 4.1, the sum converges weakly to the original image f ; if f is of bounded variation, even strong convergence is attained (cf. Theorem 4.2). In fact, in my experiments already after few decomposition steps, the original images were reconstructed (cf. Figure 6.5 and Figure 6.7). In addition, the effects characteristic for the *ROF model* still apply to the hierarchical decomposition: Especially in the first instances, potential noise is effectively removed from an image – as shown in Figure 6.6.

Since a proper choice of the initial parameter $\lambda_0 > 0$ might be difficult, I proposed a backward initialisation procedure in Section 4.3, inspired by [22]. It can be used to capture a hierarchical representation of the missing larger scales if λ_0 was chosen “too large”, meaning that u_0 is already a too detailed description of f . At least in my experiments, this concept visually worked well: The decomposition with backward initialisation yielded resultant images similar to the forward decomposition at corresponding scale, see Figure 6.9.

Bibliography

- [1] Private discussion with Prof. Dr. Jan Modersitzki. 2021.
- [2] R. Acar and C. R. Vogel. Analysis of bounded variation penalty methods for ill-posed problems. In: *Inverse problems* 10(6):1217–1229, 1994.
- [3] J. Appell, J. Banas, and N. J. M. Díaz. *Bounded variation and around*. de Gruyter, 2013.
- [4] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock. An introduction to total variation for image analysis. In: *Theoretical foundations and numerical methods for sparse recovery*. de Gruyter, 2010, pp. 263–340.
- [5] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. In: *Numerische Mathematik* 76(2):167–188, 1997.
- [6] T. Chan, S. Esedoglu, F. Park, A. Yip, et al. Recent developments in total variation image restoration. In: *Mathematical Models of Computer Vision* 17(2):17–31, 2005.
- [7] T. F. Chan and P. Mulet. On the convergence of the lagged diffusivity fixed point method in total variation image restoration. In: *SIAM journal on numerical analysis* 36(2):354–367, 1999.
- [8] L. Condat. Discrete total variation: New definition and minimization. In: *SIAM Journal on Imaging Sciences* 10(3):1258–1290, 2017.
- [9] N. Debrox, C. Le Guyader, and L. A. Vese. Multiscale Registration. In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2021, pp. 115–127.
- [10] J. Elstrodt. *Maß- und Integrationstheorie*. German. Vol. 8. Springer, 2018.
- [11] L. C. Evans and R. F. Gariepy. *Measure theory and fine properties of functions*. Chapman & Hall, 2015. ISBN: 1482242389.
- [12] P. Getreuer. Rudin-Osher-Fatemi total variation denoising using split Bregman. In: *Image Processing On Line* 2:74–95, 2012.
- [13] O. Gildemeister. *code-bachelor-thesis*. GitHub repository. 2021. URL: <https://github.com/gildol22/code-bachelor-thesis>.
- [14] M.-J. Lai, B. Lucier, and J. Wang. The convergence of a central-difference discretization of Rudin-Osher-Fatemi model for image denoising. In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2009, pp. 514–526.
- [15] *MATLAB Version 9.9.0.1538559 (R2020b)*. The Mathworks, Inc. Natick, Massachusetts, USA, 2020.
- [16] Y. Meyer. *Oscillating patterns in image processing and nonlinear evolution equations: the fifteenth Dean Jacqueline B. Lewis memorial lectures*. Vol. 22. American Mathematical Soc., 2001.
- [17] J. Modersitzki. *FAIR: flexible algorithms for image registration*. SIAM, 2009.
- [18] J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.
- [19] F. Rindler. *Calculus of variations*. Springer, 2018.
- [20] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. In: *Physica D: nonlinear phenomena* 60(1-4):259–268, 1992.
- [21] Y. Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.
- [22] E. Tadmor, S. Nezzar, and L. Vese. A multiscale image representation using hierarchical (BV, L2) decompositions. In: *Multiscale Modeling & Simulation* 2(4):554–579, 2004.
- [23] L. A. Vese and S. J. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. In: *Journal of scientific computing* 19(1):553–572, 2003.