A TGV-based framework for variational image decompression, zooming and reconstruction. Part II: Numerics

Kristian Bredies and Martin Holler*

Abstract

The present work is the second of two papers on a variational model for image reconstruction whose specific features are twofold: First, data fidelity is realized by interval constraints on the coefficients of a Riesz basis representation and second, Total Generalized Variation (TGV) of arbitrary order is employed as image prior. While the first paper provides a comprehensive analysis of the model in a general function space setting as well as discusses applications such as artifact-free JPEG, JPEG 2000 decompression and variational zooming, this second part deals with the discretized setting and presents globally convergent algorithms for these applications. In the case of JPEG and JPEG 2000 decompression, highly improved color image reconstructions are obtained from the encoded files. The numerical results are by supplemented by duality based stopping criteria, a parallelized implementation and a comparison of results obtained with Total Variation and second and third order TGV regularization.

Mathematical subject classification: 94A08 49M29 65F22 90C25
Keywords: Image reconstruction, Total Generalized Variation, JPEG decompression, JPEG 2000 decompression, Variational zooming, Primal-dual algorithm.

1 Introduction

This work is the second of two papers on a *Total Generalized Variation* (TGV) based image reconstruction model. The variational problem under consideration in the present as well as the first paper [8] reads as

$$\min_{u\in L^2(\Omega,\mathbb{R}^m)} \mathrm{TGV}_\alpha^k(u) + \mathcal{I}_{U_D}(u),$$

where TGV_α^k [9] is the vectorial Total Generalized Variation functional of order k and $\mathcal{I}_{U_D}$ is the indicator function of the convex set U_D, i.e. $\mathcal{I}_{U_D}(u) = 0$ if $u \in U_D$ and infinity else. The set U_D is given as

$$U_D = \{u \in L^2(\Omega,\mathbb{R}^m) \,|\, (Au)_n \in J_n,\ n \in \mathbb{N}\}$$

with $A : L^2(\Omega,\mathbb{R}^m) \to \ell^2$ being a Riesz-basis transformation operator and $(J_n)_n$ being closed intervals. This setting is motivated by the application to JPEG [27] decompression, where A is a

*Institute for Mathematics and Scientific Computing, Heinrichstrasse 36, A-8010 Graz, Austria. email: kristian.bredies@uni-graz.at, martin.holler@uni-graz.at.
Support by the Austrian Science Fund (FWF) special research grant SFB-F32 *Mathematical Optimization and Applications in Biomedical Sciences* is gratefully acknowledged.

block-wise cosine transform and, given a JPEG compressed file, U_D defines the set of all images that, when compressed at the given rate, result in the same file.

In the first paper [8] on this topic, the above minimization problem was investigated in a general function space setting. Existence of a solution and optimality conditions were obtained under generic assumptions. Three particular applications, namely the improved decompression of JPEG and JPEG 2000 images and a variational zooming method, were defined and shown to be concrete instances of the above setting.

The present, second part of our work is devoted to the numerical realization and practical evaluation of these three applications. After discretization, we establish a uniform algorithmic framework that is based on the primal-dual algorithm of [13] and obtain globally convergent algorithms in all three cases. We provide duality based stopping criteria that allow to estimate optimality in terms of the objective functional and an adaptive stepsize strategy that guarantees a convergent algorithm while, in practice, enabling us to overcome the stepsize constraint of [13] that yields prohibitively small stepsizes in some applications.

For JPEG and JPEG 2000 decompression, we are able to obtain highly improved reconstructions for color images by extracting the necessary information from the encoded files. In addition, for JPEG decompression, a multi-core CPU and GPU implementation is presented that obtains the improved decompression within seconds even for larger image files. As result of a general formulation, a variational zooming method based on wavelet transformations can be derived from the JPEG 2000 setting with minimal effort. While being quite simple, it yields a flexible zooming method by allowing to use any wavelet analysis operator to model down-sampling and gives a surprisingly good upsampling quality for some choices.

Both the general function space formulation as well as the numerical solution strategy allow any order of the TGV functional for regularization. Nevertheless we restrict ourselves to the second order functional in the discrete setting. This keeps the presentation of our methods simple and yields a good compromise between computational complexity and obtained image quality. We do, however, also provide experiments for JPEG decompression using the third order TGV functional for regularization.

As our framework covers different applications in mathematical image processing, there is, for each application, literature available. In particular for JPEG decompression there exist many methods to improve on the standard decompression procedure and we refer to [25, 5, 4, 1, 30] for the ones that are most related to our work. For JPEG 2000 decompression, the problem of wavelet coefficient inpainting is most related and we refer to [28, 29, 14, 24, 15] for variational approaches in this setting. Concerning image zooming, methods that approach the task by applying either a TV type regularization or aiming at wavelet inpainting for upsampling can be found in [2, 12, 22, 20, 16, 11]. In any case, we refer to [8] for a more detailed discussion on existing methods. Regarding a common framework for different applications of the above type, the only available work that goes in this direction is the TV based model [4] that preceded this work. Indeed, [4], which is based on the TV functional for regularization and an orthonormal basis for data fidelity, can be seen as motivation for the present work where non-trivial extensions have been obtained in terms of regularization, data fidelity, assumptions for existence of a solution and applications.

The present second part of our work is divided into four sections, the first one dealing with aspects that are shared by all applications of interest and the three others dealing with the particular reconstruction models.

2 Discretization and algorithmic setup

In an abstract setting, we aim at solving the following minimization problem:

$$\min_{u\in L^2(\Omega,\mathbb{R}^m)} \mathrm{TGV}_\alpha^k(u) \quad \text{subject to} \quad (a_n,u)_{L^2}\in J_n \text{ for all } n$$

with Ω a rectangular domain and $(a_n)_n$ a given Riesz basis of $L^2(\Omega,\mathbb{R}^m)$ with $(J_n)_n$ given, non-empty closed intervals, both of which depend on the concrete problem under consideration. In this section, we fix the discretization of the TGV_α^k functional and the underlying spaces. Also a stepsize adaptive primal-dual algorithm based on [13] is described in a generic form.

While we will later on present numerical results also for third order TGV regularization, we detail the discrete setting only for the second order TGV functional. The extension to the third order can be obtained with similar techniques and also the same algorithmic framework can be employed.

2.1 Discrete problem setting

Our discrete variational problem bases on functions on regular grids. The spaces of discrete multichannel images and of first and second order gradient information are defined as

$$U:=\mathbb{R}^{N\times M\times C},\quad V=U\times U,\quad W=U\times U\times U.$$

where $N,M\in\mathbb{N}$, denote the image dimensions and $C\in\mathbb{N}$ the number of (color) channels, both of which will be further specified in concrete applications.

For $u\in U$, the value $u_{i,j}^c$ is the value of the channel c of the pixel (i,j), $(u_{i,j})\in\mathbb{R}^C$ denotes the vector valued entry at the pixel (i,j) and $u^c\in\mathbb{R}^{N\times M}$ is the color component c. A norm on the entries $(u_{i,j})$ is given by

$$\big|(u_{i,j})\big|_U^2=\sum_{c=1}^C (u_{i,j}^c)^2.$$

Similarly, elements $v\in V$ and $w\in W$ are denoted by

$$v=(v_{i,j}^{\lambda,c})_{i,j,c,\lambda} \quad\text{and}\quad w=(w_{i,j}^{\lambda,c})_{i,j,c,\lambda},$$

for $0\le i<N$, $0\le j<M$, $c\in\{1,\dots,C\}$ and $\lambda\in\{1,2\}$ for v and $\lambda\in\{1,2,3\}$ for w, and norms on their entries by

$$\big|(v_{i,j})\big|_V^2=\sum_{\substack{c\in\{1,\dots,C\}\\ \lambda\in\{1,2\}}}(v_{i,j}^{\lambda,c})^2 \quad\text{and}\quad \big|(w_{i,j})\big|_W^2=\sum_{\substack{c\in\{1,\dots,C\}\\ \lambda\in\{1,2\}}}(w_{i,j}^{\lambda,c})^2+2\sum_{c=1}^C (w_{i,j}^{3,c})^2.$$

There reason for the weighing in $|\cdot|_W$ is that the three entries of w correspond to symmetric 2×2 matrices with the off diagonal entries stored only once. Standard norms and inner products on U, V and W can be obtained, for $E\in\{U,V,W\}$, $e,t\in E$, by summation over each pixel and the polarization identity, i.e.,

$$\|e\|_E^2=\sum_{\substack{0\le i<N\\ 0\le j<M}}|e_{i,j}|_E^2,\qquad (e,t)_E=\frac{1}{4}(\|e+t\|_E^2-\|e-t\|_E^2). \tag{1}$$

Besides the 2-norm, we will also need other discrete L^p norms on U, V, W. For an exponent $r \in [1,\infty]$ we always denote by $r' = r/(r-1) \in [1,\infty]$ its conjugate exponent and, abusing notation, define

$$\|e\|_r = \begin{cases} \left(\sum_{i,j} |e_{i,j}|_E^r\right)^{1/r} & \text{if } r < \infty, \\ \max_{i,j}\{|e_{i,j}|_E\} & \text{if } r = \infty, \end{cases} \tag{2}$$

for $E \in \{U, V, W\}$, $e \in E$.

The discretization of the second order TGV functional is based on its equivalent representation as detailed in [8, 7] and is given as

$$\mathrm{TGV}_\alpha^2(u) = \inf_{v \in V} \alpha_1 \|\nabla u - v\|_1 + \alpha_0 \|\mathcal{E} v\|_1, \tag{3}$$

where $\nabla : U \to V$ denotes a discrete, color component-wise gradient operator using forward differences and $\mathcal{E} : V \to W$ denotes a discrete, color component-wise symmetric gradient operator using backward differences, i.e.,

$$(\nabla u)^{1,c} = (\delta_{x+} u^c), \quad (\nabla u)^{2,c} = (\delta_{y+} u^c)$$

for $u \in U$, $c = 1, \ldots, C$, with

$$\begin{aligned} &\delta_{x+}, \delta_{y+} : \mathbb{R}^{N \times M} \to \mathbb{R}^{N \times M}, \\ (\delta_{x+} z)_{i,j} &= \begin{cases} (z_{i+1,j} - z_{i,j}) & \text{if} \quad 0 \leq i < N-1, \\ 0 & \text{if} \quad i = N-1, \end{cases} \\ (\delta_{y+} z)_{i,j} &= \begin{cases} (z_{i,j+1} - z_{i,j}) & \text{if} \quad 0 \leq j < M-1, \\ 0 & \text{if} \quad j = M-1, \end{cases} \end{aligned} \tag{4}$$

and

$$(\mathcal{E} v)^{1,c} = (\delta_{x-} v^{1,c}), \quad (\mathcal{E} v)^{2,c} = (\delta_{y-} v^{2,c}), \quad (\mathcal{E} v)^{3,c} = (\frac{\delta_{y-} v^{1,c} + \delta_{x-} v^{2,c}}{2})$$

for $v \in V$, $c = 1, \ldots, C$, with

$$\begin{aligned} &\delta_{x-}, \delta_{y-} : \mathbb{R}^{N \times M} \to \mathbb{R}^{N \times M}, \\ (\delta_{x-} z)_{i,j} &= \begin{cases} (z_{i,j} - z_{i-1,j}) & \text{if} \quad 0 < i \leq N-1, \\ 0 & \text{if} \quad i = 0, \end{cases} \\ (\delta_{y-} z)_{i,j} &= \begin{cases} (z_{i,j} - z_{i,j-1}) & \text{if} \quad 0 < j \leq M-1, \\ 0 & \text{if} \quad j = 0. \end{cases} \end{aligned} \tag{5}$$

Note that usage of forward and backward differences for the gradient ∇ and the symmetrized gradient $\mathcal{E}$, respectively, results in second order central differences for $\mathcal{E} \circ \nabla$.

We will further need discrete divergence operators which are, abusing notation, defined as $\operatorname{div} = -\nabla^*$ and $\operatorname{div} = -\mathcal{E}^*$, or, more explicitly,

$$(\operatorname{div} v)^c = -(\delta_{x+}^* v^{1,c} + \delta_{y+}^* v^{2,c})$$

and

$$(\operatorname{div} w)^{1,c} = -(\delta_{x-}^* w^{1,c} + \delta_{y-}^* w^{3,c}), \quad (\operatorname{div} w)^{2,c} = -(\delta_{x-}^* w^{3,c} + \delta_{y-}^* w^{2,c}),$$

with

$$(\delta_{x+}^* z)_{i,j} = \begin{cases} (z_{i-1,j} - z_{i,j}) & \text{if} \quad 0 < i < N-1, \\ -z_{i,j} & \text{if} \quad i = 0, \\ z_{i-1,j} & \text{if} \quad i = N-1, \end{cases} \tag{6}$$

Algorithm 1 Abstract primal-dual algorithm

```
• Initialization: Choose $\tau, \sigma > 0$ such that $\|K\|^2\tau\sigma < 1$, $(x^0, y^0) \in X \times Y$ and set $\overline{x}^0 = x^0$
• Iterations ($n \geq 0$): Update $x^n, y^n, \overline{x}^n$ as follows:
    $\begin{cases} y^{n+1} = & (I + \sigma\partial F^*)^{-1}(y^n + \sigma K\overline{x}^n) \\ x^{n+1} = & (I + \tau\partial G)^{-1}(x^n - \tau K^* y^{n+1}) \\ \overline{x}^{n+1} = & 2x^{n+1} - x^n \end{cases}$
```

and

$$(\delta_{x-}^* z)_{i,j} = \begin{cases} (z_{i,j} - z_{i+1,j}) & \text{if} \quad 0 < i < N-1, \\ -z_{i+1,j} & \text{if} \quad i = 0, \\ z_{i,j} & \text{if} \quad i = N-1, \end{cases} \tag{7}$$

and δ_{y+}^*, δ_{y-}^* defined accordingly.

2.2 A stepsize-adaptive primal-dual algorithm

The solution strategies for the sub-sequent applications are derived from the primal-dual algorithm presented in [13]. For the sake of completeness, we give a short overview of the abstract framework. Given the problem setting

$$\min_{x \in X} F(Kx) + G(x)$$

with $K : X \to Y$ a linear operator, X, Y finite dimensional Hilbert spaces, and F and G convex and l.s.c. functions, if Fenchel-Rockafellar duality is applicable, a saddle-point formulation of the above minimization problem is given as

$$\min_{x \in \mathrm{dom}(G)} \max_{y \in \mathrm{dom}(F^*)} (Kx, y)_Y + G(x) - F^*(y),$$

with F^* being the convex conjugate of F. The solution strategy now essentially consists of alternatingly performing gradient descent and ascent steps in the primal and dual direction, respectively, resulting in Algorithm 1.

There, the expression $(I + \sigma\partial F^*)^{-1}$ and $(I + \tau\partial G)^{-1}$ are the proximal maps of F^* and G, which are well-defined for any proper, convex, l.s.c. function. In our applications we employ the abstract primal-dual algorithm to suitable reformulations of the problems of interest with the aim of obtaining proximal maps that can be calculated explicitly and fast.

As shown in [13] the algorithm is globally convergent if the stepsizes σ, τ satisfy $\sigma\tau\|K\|^2 < 1$. An estimate on $\|K\|$, however, has to include all pathological cases and thus the stepsize restriction can be too conservative in some applications. On the other hand, we can learn from the proof of convergence for Algorithm 1 as given in [13, Theorem 1] that this restriction is in fact only needed to ensure that

$$\|K(x^n - x^{n-1})\|_Y < \frac{1}{\sqrt{\sigma\tau}}\|x^n - x^{n-1}\|_X \tag{8}$$

is satisfied in each iteration. This means that one might only use the iterates to estimate $\|K\|$ and doing so, potentially obtain convergence for larger stepsizes. Motivated by this, we formulate the following criterion for adaptive stepsize choice, that still allows to ensure convergence of the primal-dual algorithm:

Given $\theta \in (0,1)$ and $\eta > 0$, after each iteration of Algorithm 1, update σ and τ as

$$\sigma = \mathcal{S}\left(\sigma\tau, \frac{\|x^{n+1} - x^n\|_X}{\|K(x^{n+1} - x^n)\|_Y}\right)\eta, \qquad \tau = \mathcal{S}\left(\sigma\tau, \frac{\|x^{n+1} - x^n\|_X}{\|K(x^{n+1} - x^n)\|_Y}\right)/\eta$$

with

$$\mathcal{S}(\sigma\tau, \mathfrak{n}) = \begin{cases} \mathfrak{n} & \text{if } \theta\sigma\tau \geq \mathfrak{n}, \\ \sqrt{\theta\sigma\tau} & \text{if } \sigma\tau \geq \mathfrak{n} > \theta\sigma\tau, \\ \sqrt{\sigma\tau} & \text{else.} \end{cases} \tag{9}$$

As each reduction of σ and τ is done at least by a constant factor $\sqrt{\theta} < 1$, this rule guarantees that, after finitely many reductions, equation (8) holds for all remaining iterates. In practice, this situation is often already reached for stepsizes that still cannot be guaranteed to satisfy $\sigma\tau L^2 < 1$ with L being an analytical upper bound of $\|K\|$ and consequently, as we will see for example in the application to JPEG 2000 decompression, convergence can be obtained using significantly larger stepsizes. Note that the choice of η allows to balance between the primal stepsize τ and the dual stepsize σ. As we did, for the applications considered in this work, not observe a significant improvement with unequal primal and dual stepsizes, we choose $\eta = 1$ in practice.

3 Color JPEG decompression

3.1 Discrete setting

Given the minimization problem [8, Equation (22)] for artifact free JPEG decompression, defined in function space setting, we will now handle its discrete equivalent and show how a solution can be obtained numerically. For that purpose, we use the notation as given in Section 2. We note that some discrete formulations for this problem have already been presented in [5], however, a more extensive treatment is presented in the following. We assume that the horizontal and vertical number of pixels of the discrete images are multiples of 8 and consider three color components, i.e., U is given by $U = \mathbb{R}^{8k \times 8l \times 3}$, $k, l \in \mathbb{N}$. In order to describe the set of all images matching a given, JPEG compressed file, we need discrete versions of a subsampling and a block-wise cosine transform operator S and C, respectively, for which we only give a local, component-wise definition in order to avoid extensive indexing.

A subsampling operator $S : U \to \tilde{U}$, with $\tilde{U} = \mathbb{R}^{\tilde{N}_1 \times \tilde{M}_1} \times \mathbb{R}^{\tilde{N}_2 \times \tilde{M}_2} \times \mathbb{R}^{\tilde{N}_3 \times \tilde{M}_3}$ being the space of subsampled images, is defined component-wise, locally, for subsampling factors f_1, f_2 of the horizontal and vertical directions and $(r_{i,j})_{i,j}$ with $0 \leq i < f_1$, $0 \leq j < f_2$, as

$$Sr = \frac{1}{f_1 f_2} \sum_{m=0}^{f_1-1} \sum_{n=0}^{f_2-1} r_{m,n}, \tag{10}$$

reducing the resolution of the component by factors f_1 and f_2 in the vertical and horizontal direction, respectively. If the resolution of one component is not reduced, as typically for the brightness component, S is the identity for this component. The discrete cosine transformation operator is defined, for each color component, on each disjoint 8×8 block $(r_{i,j})_{0 \leq i,j < 8}$, as

$$(Cr)_{p,q} = c_p c_q \sum_{n,m=0}^{7} r_{n,m} \cos\left(\frac{\pi(2n+1)p}{16}\right) \cos\left(\frac{\pi(2m+1)q}{16}\right), \tag{11}$$

for $0 \leq p, q < 8$ and

$$c_s = \begin{cases} \frac{1}{\sqrt{8}} & \text{if } s = 0, \\ \frac{1}{2} & \text{if } 1 \leq s < 8. \end{cases}$$

Having this, the discrete data set U_D of all images that would result in the same, given JPEG file, can now be described by

$$U_D = \{u \in U \,|\, CSu \in D\}, \tag{12}$$

with $D = \{d \,|\, d^c_{i,j} \in J^c_{i,j}\}$ and $J^c_{i,j} = [l^c_{i,j}, o^c_{i,j}]$ data intervals that can be obtained from the quantized DCT coefficients stored in the compressed JPEG file as in [8, Equation (16)] (see also [4, Section 5]).

With these prerequisites, the finite dimensional optimization problem for artifact-free JPEG decompression reads as

$$\min_{u \in U} \mathrm{TGV}^2_\alpha(u) + \mathcal{I}_{U_D}(u). \tag{13}$$

with

$$\mathcal{I}_{U_D}(u) = \begin{cases} 0 & \text{if } u \in U_D, \\ \infty & \text{else.} \end{cases}$$

Now in order to solve (13) numerically with the primal-dual algorithm described in Subsection 2.2, we formulate an equivalent saddle-point problem. For that purpose, we define X and Y as

$$X := U \times V, \quad Y := V \times W,$$

equipped with the norms

$$\|(u, v)\|^2_X = \|u\|^2_U + \|v\|^2_V, \quad \|(p, q)\|^2_Y = \|p\|^2_V + \|q\|^2_W,$$

resulting from inner products $(\cdot, \cdot)_X$, $(\cdot, \cdot)_Y$, the functions $F : Y \to \mathbb{R}$, $G : X \to \mathbb{R}$,

$$F(v, w) = \alpha_1 \|v\|_1 + \alpha_0 \|w\|_1, \quad G(u, v) = \mathcal{I}_{U_D}(u) \tag{14}$$

and the operator $K : X \to Y$,

$$K = \begin{bmatrix} \nabla & -\mathrm{I} \\ 0 & \mathcal{E} \end{bmatrix}, \tag{15}$$

with $\mathrm{I} : V \to V$ denoting the identity. A saddle-point formulation of Problem (13) can now be obtained as follows.

Proposition 3.1. *The discrete minimization problem* (13) *related to artifact-free JPEG decompression possesses a solution and is equivalent to*

$$\min_{x \in X} F(Kx) + G(x). \tag{16}$$

Further there exists a solution to the dual problem of (16)*, characterized by*

$$\max_{y \in Y} -G^*(-K^*y) - F^*(y), \tag{17}$$

as well as to the saddle point problem, given by

$$\min_{x \in dom(G)} \max_{y \in dom(F^*)} (Kx, y)_Y - F^*(y) + G(x). \tag{18}$$

In addition, $\hat{x}, \hat{y}$ are solutions to the primal and dual problem, respectively, if and only if $(\hat{x}, \hat{y})$ solves the saddle point problem.

Proof. Existence of a solution to (13) can be obtained similar as in the continuous setting (see [8]) and the equivalence to the primal problem (16) is obvious. Given that F is continuous, existence of a solution to the dual problem as well as equality of the primal and the dual problem at optimal points follow immediately from [18, Theorem III.4.1]. The equivalence of the saddle point problem then follows from [18, Proposition III.3.1]. □

Note that G^* and F^* are given by

$$G^*(u^*,v^*) = \sup_{u \in U_D} (u^*,u)_U + \mathcal{I}_{\{0\}}(v^*),$$

where

$$\mathcal{I}_{\{0\}}(v) = \begin{cases} 0 & \text{if } v = 0, \\ \infty & \text{else,} \end{cases}$$

and

$$F^*(p,q) = \mathcal{I}_{\{\|\cdot\|_\infty \leq \alpha_1\}}(p) + \mathcal{I}_{\{\|\cdot\|_\infty \leq \alpha_0\}}(q),$$

where

$$\mathcal{I}_{\{\|\cdot\|_\infty \leq \lambda\}}(e) = \begin{cases} 0 & \text{if } \|e\|_\infty \leq \lambda, \\ \infty & \text{else.} \end{cases}$$

Further, the operator K^* denotes the adjoint of K and is given by

$$K^* = \begin{bmatrix} -\operatorname{div} & 0 \\ -\mathrm{I} & -\operatorname{div} \end{bmatrix}.$$

By using standard arguments from convex analysis, we can also obtain an optimality condition for the discrete saddle point problem that will be helpful when deriving a stopping criterion for the algorithm later on (see also [8, Theorem 3.3] for optimality conditions in the general, infinite dimensional setting).

Proposition 3.2. *There exists a solution to* (18) *and* $(\hat{x},\hat{y}) = (\hat{u},\hat{v},\hat{p},\hat{q})$ *being optimal is equivalent to*

- *For all* $0 \leq i < N$, $0 \leq j < M$,

$$\begin{cases} \hat{p}_{i,j} = \alpha_1 \dfrac{(\nabla \hat{u} - \hat{v})_{i,j}}{|(\nabla \hat{u} - \hat{v})_{i,j}|_V} & \textit{if } (\nabla \hat{u} - \hat{v})_{i,j} \neq 0 \textit{ and } |\hat{p}_{i,j}|_V \leq \alpha_0 \textit{ else,} \\ \hat{q}_{i,j} = \alpha_0 \dfrac{(\mathcal{E}\hat{v})_{i,j}}{|(\mathcal{E}\hat{v})_{i,j}|_W} & \textit{if } (\mathcal{E}\hat{v})_{i,j} \neq 0 \textit{ and } |\hat{q}_{i,j}|_W \leq \alpha_1 \textit{ else,} \end{cases}$$

- $\hat{p} + \operatorname{div} \hat{q} = 0$,
- $\hat{u} \in U_D$ *and* $\operatorname{div} \hat{p} = S^*C^*\hat{w}$ *with* $\hat{w} \in \tilde{U}$ *such that*

$$\begin{cases} \hat{w}^c_{i,j} \geq 0 & \textit{if } (CS\hat{u})^c_{i,j} = \sup(J^c_{i,j}) \neq \inf(J^c_{i,j}), \\ \hat{w}^c_{i,j} \leq 0 & \textit{if } (CS\hat{u})^c_{i,j} = \inf(J^c_{i,j}) \neq \sup(J^c_{i,j}), \\ \hat{w}^c_{i,j} = 0 & \textit{if } (CS\hat{u})^c_{i,j} \in \operatorname{int}(J^c_{i,j}). \end{cases}$$

3.2 Practical implementation

We solve the saddle point problem (18) using the primal-dual algorithm presented in Subsection 2.2. As we will see, all necessary steps during one iteration reduce to simple arithmetic operations and the evaluation of a forward and inverse blockwise cosine transformation, for which highly optimized code already exists. This makes the algorithm fast and also easy to parallelize. Also, we can obtain a suitable stopping rule ensuring optimality.

Using standard arguments from convex analysis, the proximal map $(I+\sigma\partial F^*)^{-1}$ can be given as

$$(I+\sigma\partial F^*)^{-1}(p,q) = \left(\mathrm{proj}_{\alpha_1}(p), \mathrm{proj}_{\alpha_0}(q)\right),$$

where, for $E \in \{V, W\}$, $e \in E$,

$$(\mathrm{proj}_\lambda(e))_{i,j} = \frac{e_{i,j}}{\max(1, \frac{|e_{i,j}|_E}{\lambda})}. \tag{19}$$

Similar, the evaluation of $(I+\tau\partial G)^{-1}$ is the identity for v and the projection on the data set

$$U_D = \{u \in U \,|\, CSu \in D\}$$

of u in $x = (u, v)$. As the following lemma shows, in the particular setting for JPEG decompression, this projection can be reduced to a projection on

$$U_C = \{u \in \tilde{U} \,|\, Cu \in D\}$$

by

$$P_{U_D}(u) = u + \tilde{S}^*\left(P_{U_C}(Su) - Su\right),$$

where $\tilde{S}^*$ is the adjoint of a re-weighted subsampling operator $\tilde{S} : U \to \tilde{U}$, and is given locally by replication of $r \in \mathbb{R}$, i.e.,

$$\tilde{S}^* r = \begin{pmatrix} r & \dots & r \\ \vdots & & \vdots \\ r & \dots & r \end{pmatrix}.$$

Lemma 3.1. *Let H_1, H_2 be two Hilbert spaces and $B : H_1 \to H_2$ be a bounded linear operator such that, for all $v \in H_2$, $BB^* = c\,\mathrm{I}$ with $c \in (0, \infty)$. With $D \subset H_2$ a non-empty, closed, convex set and*

$$V_D = \{u \in H_1 \,|\, Bu \in D\}$$

we get that for all $\hat{u} \in H_1$,

$$P_{V_D}(\hat{u}) = \hat{u} + \frac{1}{c}B^*\left(P_D(B\hat{u}) - B\hat{u}\right)$$

with P_{V_D} and P_D being the projection operator onto V_D and D, respectively.

Proof. Take $\hat{u} \in H_1$. Then,

$$B(\hat{u} + \frac{1}{c}B^*(P_D(B\hat{u}) - B\hat{u}) = P_D(B\hat{u})) \in D,$$

and thus it is left to show that

$$(\hat{u} - \hat{u} - \frac{1}{c}B^*\left(P_D(B\hat{u}) - B\hat{u}\right), u - \hat{u} - \frac{1}{c}B^*\left(P_D(B\hat{u}) - B\hat{u}\right))_{H_1} \leq 0$$

for all $u \in V_D$. Note that we have

$$(B\hat{u} - P_D(B\hat{u}), v - P_D(B\hat{u}))_{H_2} \leq 0 \quad \text{for all } v \in D.$$

Using this we get, for all $u \in V_D$,

$$\begin{aligned}(\hat{u} - \hat{u} - \frac{1}{c}B^* \left(P_D(B\hat{u}) - B\hat{u}\right), &u - \hat{u} - \frac{1}{c}B^* \left(P_D(B\hat{u}) - B\hat{u}\right))_{H_1} \\ &= \frac{1}{c}(B^* \left(B\hat{u} - P_D(B\hat{u})\right), u - \hat{u} - \frac{1}{c}B^* \left(P_D(B\hat{u}) - B\hat{u}\right))_{H_1} \\ &= \frac{1}{c}(B\hat{u} - P_D(B\hat{u}), Bu - P_D(B\hat{u}))_{H_2} \leq 0.\end{aligned}$$ □

Orthogonality of C then allows to reduce $(I + \sigma\partial G)^{-1}$ to

$$(I + \tau\partial G)^{-1}(u, v) = (u + \tilde{S}^*\big(\mathrm{proj}_{U_C}(Su) - Su\big), v),$$

where

$$\mathrm{proj}_{U_C}(u) = C^* w,$$

with

$$w^c_{i,j} = \begin{cases} u^c_{i,j} & \text{if } (Cu)^c_{i,j} \in J^c_{i,j}, \\ \sup(J^c_{i,j}) & \text{if } (Cu)^c_{i,j} > \sup(J^c_{i,j}), \\ \inf(J^c_{i,j}) & \text{if } (Cu)^c_{i,j} < \inf(J^c_{i,j}). \end{cases}$$

Altogether, the concrete implementation of the primal-dual algorithm for JPEG decompression can be given in Algorithm 2. Note that the step-size restriction $\sigma\tau \leq \frac{1}{12}$ results from combining the estimates $\|\nabla\| \leq \sqrt{8}$, $\|\mathcal{E}\| \leq \sqrt{8}$ to get $\|K\|^2 < 12$ (see [10]). As one can see, all steps of Algorithm 2 can be evaluated by simple, mostly pixel-wise operations making each iteration step fast.

Stopping criterion

In order to validate our numerical solution, we seek for a suitable stopping rule. Our goal is to estimate optimality of the iterates in terms of the objective function. The basis for that is the following observation, which holds for any optimal solution of the primal problem (16) $\hat{x}$ and any $x \in X$, $y \in Y$.

$$\begin{aligned} 0 \leq F(Kx) - F(K\hat{x}) &= F(Kx) - \left(\sup_{z \in Y}(\hat{x}, K^*z)_X - F^*(z)\right) \\ &\leq F(Kx) - (\hat{x}, K^*y)_X + F^*(y). \end{aligned} \tag{20}$$

As the following proposition shows, the last term of the above inequalities can be suitably estimated using the iterates of Algorithm 2.

Proposition 3.3. *Take $1 < \gamma$, $1 \leq r \leq \infty$, $(x_n, y_n)_n = ((u_n, v_n, p_n, q_n))_n$ to be the iterates of Algorithm 2 and $(\hat{x}, \hat{y}) = (\hat{u}, \hat{v}, \hat{p}, \hat{q})$ a saddle point of (18). Denoting $J^c_{i,j} = [l^c_{i,j}, o^c_{i,j}]$, define*

$$\begin{aligned}\mathcal{G}(x_n, y_n) = F(Kx_n) &+ T_n\| \operatorname{div}^2 \tilde{q}_n - S^*\tilde{S} \operatorname{div}^2 \tilde{q}_n\|_{r'} \\ &+ \sum_{i,j,c}\left(\frac{o^c_{i,j} + l^c_{i,j}}{2}(-C\tilde{S} \operatorname{div}^2 \tilde{q}_n)^c_{i,j} + \frac{o^c_{i,j} - l^c_{i,j}}{2}|(-C\tilde{S} \operatorname{div}^2 \tilde{q}_n)^c_{i,j}|\right),\end{aligned} \tag{21}$$

Algorithm 2 Scheme of implementation for JPEG decompression

```
function TGV-JPEG($J_{\mathrm{comp}}$)
    $(J^c_{i,j})_{i,j,c} \leftarrow$ Decoding of JPEG-Object $J_{\mathrm{comp}}$
    $d \leftarrow$ Midpoints of $(J^c_{i,j})_{i,j,c}$
    $u \leftarrow \tilde{S}^* C^* d$
    $v \leftarrow 0, \overline{u} \leftarrow u, \overline{v} \leftarrow 0, p \leftarrow 0, q \leftarrow 0$
    choose $\sigma, \tau > 0$ such that $\sigma\tau \leq 1/12$
    repeat
        $p \leftarrow \mathrm{proj}_{\alpha_1}(p + \sigma(\nabla \overline{u} - \overline{v}))$
        $q \leftarrow \mathrm{proj}_{\alpha_0}(q + \sigma(\mathcal{E}\overline{v})$
        $u_+ \leftarrow u + \tau(\operatorname{div} p)$
        $v_+ \leftarrow v + \tau(p + \operatorname{div} q)$
        $u_+ \leftarrow u_+ + \tilde{S}^*(\mathrm{proj}_{U_C}(Su_+) - Su_+)$
        $\overline{u} \leftarrow (2u_+ - u)$, $\overline{v} \leftarrow (2v_+ - v)$
        $u \leftarrow u_+$, $v \leftarrow v_+$
    until Stopping criterion fulfilled
    return $u_+$
end function
```

where $\tilde{q}_n := \beta_n q_n$ *with* $\beta_n := \frac{\alpha_1}{\max(\alpha_1, \|\operatorname{div} q_n\|_\infty)}$, *and* $T_n = \gamma\|u_n - \tilde{S}^* S u_n\|_r$. *Then,*

$$\mathcal{G}(x_n, y_n) \to 0 \text{ as } n \to \infty$$

and, additionally,

$$\mathcal{G}(x_n, y_n) \geq F(Kx_n) - \mathrm{TGV}^2_\alpha(\hat{u}) \geq 0 \tag{22}$$

whenever $T_n \geq \|\hat{u} - \tilde{S}^* S\hat{u}\|_r$ *which is eventually the case.*

Proof. At first we show convergence of $\mathcal{G}(x_n, y_n)$ to zero, which follows from convergence of the iterates, the optimality conditions given in Proposition 3.2 and strong duality as shown in the proof of Proposition 3.1. Indeed, as $\beta_n \to 1$ and

$$\operatorname{div}^2 \tilde{q}_n = \operatorname{div}^2 \beta_n q_n \to \operatorname{div}^2 \hat{q} = -\operatorname{div} \hat{p} = -S^* C^* \hat{w}, \quad \text{for } n \to \infty,$$

and some $\hat{w} \in \tilde{U}$, it follows from this identity as well as $\tilde{S}S^* = \mathrm{I}$ that $\|\operatorname{div}^2 \tilde{q}_n - S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n\|_{r'} \to 0$ as $n \to \infty$.

Now, a straightforward, pointwise calculation shows that, for $w \in \tilde{U}$,

$$\sup_{d \in D} (d, w)_{\tilde{U}} = \sum_{i,j,c} \left(\frac{o^c_{i,j} + l^c_{i,j}}{2} w^c_{i,j} + \frac{o^c_{i,j} - l^c_{i,j}}{2} |w^c_{i,j}| \right). \tag{23}$$

This implies in particular that $w \mapsto \sup_{d \in D}(d, w)_{\tilde{U}}$ is continuous and, using the optimality conditions of Proposition 3.2, we hence get, for $n \to \infty$,

$$\sup_{d \in D}(d, -C\tilde{S} \operatorname{div}^2 \tilde{q}_n)_{\tilde{U}} \to \sup_{d \in D}(d, \hat{w})_{\tilde{U}} = \sup_{u \in U_D} (u, S^* C^* \hat{w})_U = \sup_{u \in U_D} (u, \operatorname{div} \hat{p})_U = G^*(-K^*(\hat{p}, \hat{q})).$$

Continuity of $x \mapsto F(Kx)$ further implies that

$$F(Kx_n) \to F(K\hat{x})$$

and hence, by strong duality, it follows that, for $n \to \infty$,

$$\begin{aligned} \mathcal{G}(x_n, y_n) &= F(Kx_n) + T_n \| \operatorname{div}^2 \tilde{q}_n - S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n \|_{r'} + \sup_{d \in D} (d, -C\tilde{S} \operatorname{div}^2 \tilde{q}_n)_{\tilde{U}} \\ &\to F(K\hat{x}) + G^*(-K^*\hat{y}) = F(K\hat{x}) + G(\hat{x}) - (-G^*(-K^*\hat{y}) - F^*(\hat{y})) = 0. \end{aligned}$$

In order to get the estimate (22) we insert $x_n = (u_n, v_n)$, $\tilde{y}_n = (-\operatorname{div} \tilde{q}_n, \tilde{q}_n)$ in (20). Since, due to our choice of β_n and the projections in Algorithm 2, $\| \operatorname{div} \tilde{q}_n \|_\infty \leq \alpha_1$ as well as $\| \tilde{q}_n \|_\infty \leq \alpha_0$, the last term in (20) is given as

$$F(Kx_n) - (\hat{x}, K^*\tilde{y}_n)_X + F^*(\tilde{y}_n) = F(Kx_n) + (\hat{u}, -\operatorname{div}^2 \tilde{q}_n)_U. \tag{24}$$

Using that both $\tilde{S}S^*$ and $S\tilde{S}^*$ are the identity we can estimate

$$\begin{aligned} & (\hat{u}, -\operatorname{div}^2 \tilde{q}_n)_U \\ = & \Big[(\hat{u} - \tilde{S}^* S \hat{u} + \tilde{S}^* S \hat{u}, -\operatorname{div}^2 \tilde{q}_n + S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n - S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n)_U \Big] \\ = & \Big[(\hat{u} - \tilde{S}^* S \hat{u}, -\operatorname{div}^2 \tilde{q}_n + S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n)_U + (\tilde{S}^* S \hat{u}, -S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n)_U \Big] \\ \leq & \| \hat{u} - \tilde{S}^* S \hat{u} \|_r \| \operatorname{div}^2 \tilde{q}_n - S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n \|_{r'} + (S\hat{u}, -\tilde{S} \operatorname{div}^2 \tilde{q}_n)_U \\ \leq & \| \hat{u} - \tilde{S}^* S \hat{u} \|_r \| \operatorname{div}^2 \tilde{q}_n - S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n \|_{r'} + \sup_{u \in U_D} (Su, -\tilde{S} \operatorname{div}^2 \tilde{q}_n)_U \\ = & \| \hat{u} - \tilde{S}^* S \hat{u} \|_r \| \operatorname{div}^2 \tilde{q}_n - S^* \tilde{S} \operatorname{div}^2 \tilde{q}_n \|_{r'} + \sup_{v \in D} (v, -C\tilde{S} \operatorname{div}^2 \tilde{q}_n)_{\tilde{U}}, \end{aligned} \tag{25}$$

where the supremum in the last expression is given as in (23). Combining this with the estimates in (25), (24) and (20) and assuming that $\| \hat{u} - \tilde{S}^* S \hat{u} \|_r \leq T_n$ the estimate (22) follows. At last, we note that, since $\gamma > 1$ and $u_n \to \hat{u}$ as $n \to \infty$, $\| \hat{u} - \tilde{S}^* S \hat{u} \|_r \leq T_n$ is satisfied after a finite number of iterations. □

This allows, for given $\epsilon > 0$, to use $\mathcal{G}(x_n, y_n) < \epsilon$ as stopping criterion. Note also that, if no supsampling is performed, i.e., S is the identity, the estimate (22) holds without any assumptions on T_n.

3.3 Numerical experiments

We now present a numerical evaluation of the proposed method for artifact free JPEG decompression. To allow for an additional numerical evaluation beyond the results presented in this section, we included the source code of our decompression algorithm in the supplementary material of the paper (see also `http://www.uni-graz.at/~hollerm`). We tested the method for several lossy JPEG compressed images, where the memory requirement of each JPEG compressed image (including lossless compression) is given in *bits per pixel (bpp)*. Note that we consider 8 bit grayscale- and 24 bit true color images, i.e. an uncompressed grayscale and color image requires 8 and 24 bpp, respectively.

For decompression, we fix the ratio between α_0 and α_1 for the evaluation of the TGV_α^2 functional (cf. equation (3)) as $\frac{\alpha_0}{\alpha_1} = \sqrt{2}$ based on empirical observations. As stopping criterion we use either the iteration number, if this is necessary for comparability, or a normalized version of the modified primal-dual gap $\mathcal{G}$ introduced in (21), which is given by

$$\overline{\mathcal{G}}(x_n, y_n) = \frac{1}{8k8l}\mathcal{G}(x_n, y_n) \tag{26}$$

with (x_n, y_n) the current iterates of Algorithm 2. The parameters in the definition of $\mathcal{G}$ are fixed to $r = r' = 2$, as this resulted in the lowest values of $\mathcal{G}$ in practice, and $\gamma = 1.001$. We point out that, while $\gamma > 1$ is required by Proposition 3.3, we observed that the estimate (22) is in practice satisfied even for $\gamma = 0$. This is, however, not the case in the application to JPEG 2000 decompression and we ask for the readers patience until Section 4.3 for a more detailed discussion. The reason for normalizing the modified primal-dual gap is to make it image size independent and, taking into account equation (22), to get an estimate on the average pixel-wise difference

$$\left[|(\nabla u_n - v_n)_{i,j}|_V + |(\mathcal{E}v_n)_{i,j}|_W\right] - \left[|(\nabla \hat{u} - \hat{v})_{i,j}|_V + |(\mathcal{E}\hat{v})_{i,j}|_W\right],$$

where again $(\hat{u}, \hat{v})$ denote solutions of the primal problem (16). Considering this estimate, it is important to note that in the numerical computations we process our images within the range [0,255], thus a pixel wise error of, e.g., less than one results in a reconstruction that is almost not distinguishable from the optimal solution.

As first example, Figure 1 depicts the standard decompression of two JPEG compressed images, corrupted by blocking and ringing artifacts, and their improved decompression obtained with our TGV based method. As stopping criterion we require the normalized modified primal-dual gap to be below $\epsilon = 10^{-1}$, which resulted in 1668 iterations for the top and 1139 iterations for the bottom image. As one can see, the blocking and ringing artifacts appearing in the standard decompression have been removed in the improved version, while edges are kept sharp, leading to more natural and visually more appealing images.

Next, Figure 3.3, compares different orders of TGV regularization on a synthetic test image. The original, uncompressed image and a surface plot of the region marked in red is shown on the top. The image was then compressed to 0.56 bpp. The lines below depict, from top to bottom, results obtained with the standard decompression method, a TV, TGV_α^2 and TGV_α^3 regularized decompression. Each line shows a magnification of the region marked in blue (left), surface plots of the region marked in red (middle), and a difference images in logarithmic scale (right). The TV and TGV_α^3 regularized decompressions were obtained with a similar implementation as presented in Subsection 3.2 for second order TGV regularization. To ensure optimality, 5000 iterations were performed with each method.

The surface plots clearly show artifacts in the standard decompressed version, but also a staircasing in the TV regularized result. For second and third order TGV regularization, only marginal differences are visible in the image sections on the left and the surface plot in the middle. The difference images, however, still show a significant improvement in the step from second to third order TGV regularization.

3.4 A GPU implementation

As already discussed in [5], we also developed a parallel implementation of the reconstruction method for multi-core CPUs and GPUs, using OpenMP [26] and Nvidia's Cuda [23], respectively. For the GPU implementation we partly used kernel functions adapted to the compute capability of the device. The blockwise DCT was performed on the CPU and the GPU using FFTW [19] and a block-DCT kernel provided by the Cuda SDK, respectively. Computation times of those implementations for multiple image sizes are given in Table 1, taken from [5]. The relative time cost of particular iteration steps is compared in Table 2. As one can see, especially the GPU implementation yields a high acceleration and makes the method suitable for practical applications. The given computation times correspond to the computation of 1000 iterations,

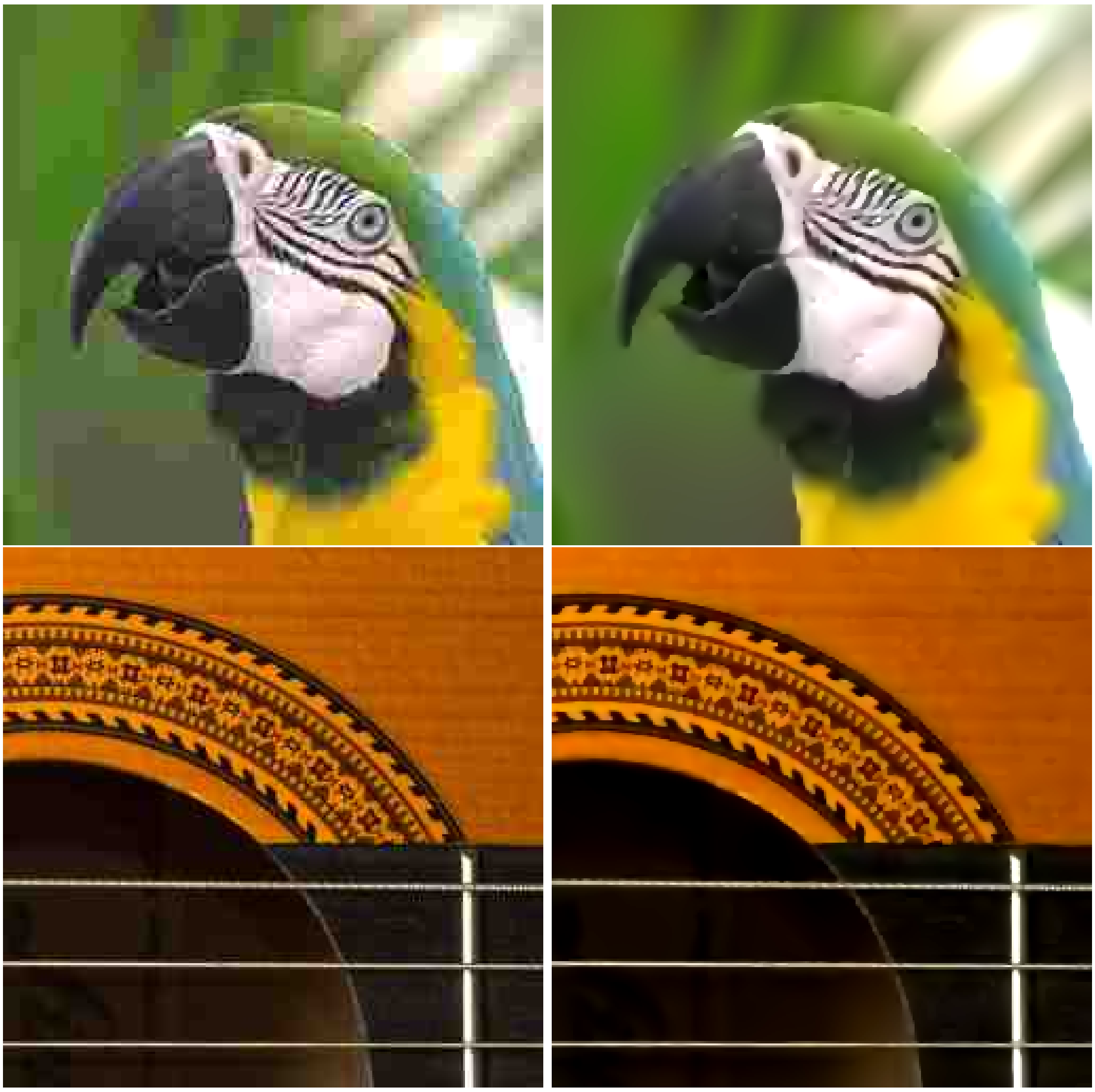

Figure 1: On the left: Standard decompression. On the right: TGV-based reconstruction obtained with normalized modified primal-dual gap below 10^{-1} as stopping criterion. Top: Parrot image at 0.3 bpp (256×256 pixels, 1668 iterations). Bottom: Guitar image at 1.06 bpp (256×256 pixels, 1139 iterations)

which is motivated by the number of iterations resulting from a normalized modified primal-dual gap below 10^{-1} as stopping criterion. Let us remark however that, since the decrease of the TGV-value of the image is typically very high especially during the first iterations of the algorithm, and since $u_n \in U_D$ can be ensured for any iteration step image u_n, one can also use the images obtained after only a few iterations as (intermediate or final) reconstruction. This yields a practicable method that allows to improve given JPEG images in almost real time.

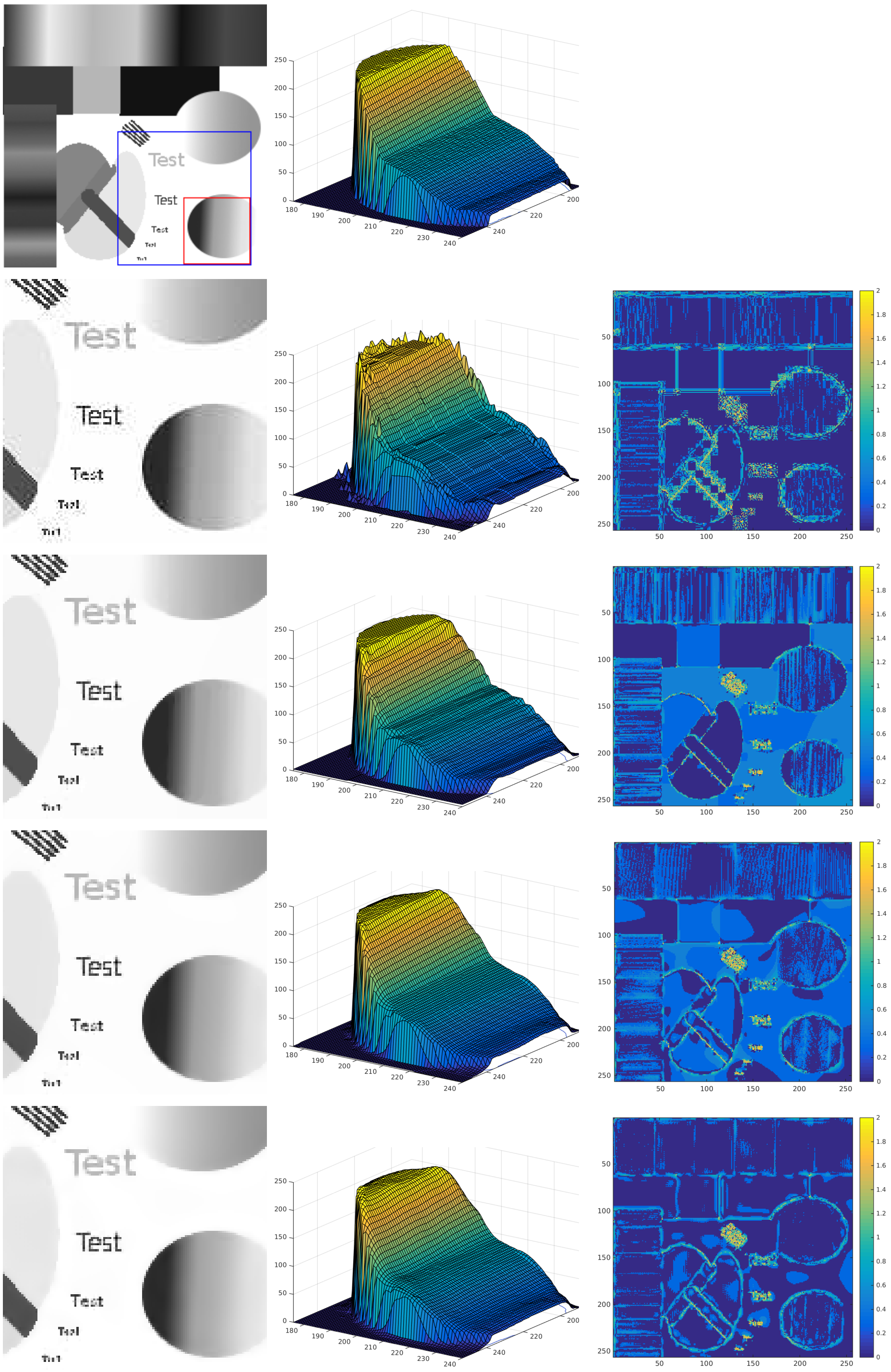

Figure 2: Top: Original image with red region plotted as surface. Below: Blue section (left), surface plot (center) and difference image (right) for different decompression methods at 0.56 bpp. Top to bottom: Standard, TV, TGV^2_α, TGV^3_α regularized decompression.

Table 1: Computation times in seconds to perform 1000 iterations for different devices and image sizes. CPU: AMD Phenom 9950. GPUs: Nvidia Quadro FX 3700 (compute capability 1.1), Nvidia GTX 280 (compute capability 1.3), Nvidia GTX 580 (compute capability 2.0). Note that on the Quadro FX 3700 and GTX 280, not enough memory was available to perform the algorithm for the 3200 × 2400 pixel image.

Device	512 × 512	1600 × 1200	3200 × 2400
CPU Single-core	53.22	672.51	1613.44
CPU Quad-core	28.32	263.70	812.18
GPU Quadro FX 3700	4.92	35.52	-
GPU Nvidia GTX 280	2.2	10.22	-
GPU Nvidia GTX 580	1.2	6.6	25.70

Table 2: Relative computation times in percent for different iteration steps. The data was obtained by reconstructing a 1600 × 1200 pixel image with 1000 iterations.

Iteration step	CPU	GPU
$\left\{\begin{array}{l} p \leftarrow \operatorname{proj}_{\alpha_1}(p + \sigma(\nabla\overline{u} - \overline{v})) \\ q \leftarrow \operatorname{proj}_{\alpha_0}(q + \sigma(\mathcal{E}(\overline{v})) \end{array}\right\}$	52%	32%
$\left\{\begin{array}{l} u_+ \leftarrow u + \tau(\operatorname{div} p) \\ v_+ \leftarrow v + \tau(p + \operatorname{div} q) \end{array}\right\}$	19%	29%
$\left\{ u_+ \leftarrow u_+ + \tilde{S}^*(\operatorname{proj}_{U_C}(Su_+) - Su_+) \right\}$	20%	21%
$\left\{\begin{array}{l} \overline{u} \leftarrow (2u_+ - u) \\ \overline{v} \leftarrow (2v_+ - v) \end{array}\right\}$	9%	18%

4 Color JPEG 2000 decompression

4.1 Discrete setting

In this section we deal with the discrete implementation of the decompression model for JPEG 2000 images as described in [8, Subsection 4.3]. Essentially, the lossy part of JPEG 2000 compression is based on a biorthogonal wavelet transform using either CDF 9/7 or Le Gall 5/3 wavelets, followed by quantization of the resulting coefficients. As a consequence, the set of all images whose compression results in the same, given file will be defined by imposing constraints on the wavelet coefficients. However, as the quantization procedure of JPEG 2000 is more involved than the one of JPEG, it will not be possible to obtain finite bounds for all coefficients.

For the sake of simplicity, we consider quadratic images and hence assume the finite dimensional space of discrete color images to be given as $U = \mathbb{R}^{N\times N\times 3}$, with $N \in \mathbb{N}$. As described in [8, Subsection 4.3], for any given resolution level $R \in \mathbb{Z}$, we can obtain two dual Riesz bases of $L^2((0,1) \times (0,1))$, denoted by

$$(\Phi_{R,k})_k \cup (\Psi_{j,k})_{j,k} \quad \text{and} \quad (\tilde{\Phi}_{R,k})_k \cup (\tilde{\Psi}_{j,k})_{j,k}, \tag{27}$$

that correspond to either CDF 9/7 or Le Gall 5/3 wavelets. Possibly shifting resolution levels, we assume that $(\Phi_{0,k})_k$, for $0 \leq k \leq N^2 - 1$ constitutes a basis of $\mathbb{R}^{N\times N}$ and hence, each color component $u^c \in \mathbb{R}^{N\times N}$of $u \in U$ is uniquely defined by the inner products

$$(u^c,\, \Phi_{0,k}) \quad \text{for } 0 \leq k < N^2,\, c \in \{1,2,3\}.$$

For a given JPEG 2000 compressed file, we can then obtain a resolution level $R \in \mathbb{N}$ and, for each color component $c \in \{1,2,3\}$, data intervals

$$(J^{\Phi,c}_{R,k})_k,\, (J^{\Psi,c}_{j,k})_{j,k}, \tag{28}$$

such that the source image $u \in U$ of the compression process satisfies

$$(u^c, \Phi_{R,k}) \in J^{\Phi,c}_{R,k}, \quad (u^c, \Psi_{j,k}) \in J^{\Psi,c}_{j,k}, \tag{29}$$

for all j,k and $c \in \{1,2,3\}$. Due to the multi-resolution framework of the wavelet basis, the $J^{\Phi,c}_{R,k}$ and $J^{\Psi,c}_{j,k}$ provide the low resolution and detail information, respectively, about the image to decompress. As result of the JPEG 2000 coding, each of these intervals is either bounded from both sides or contains all of $\mathbb{R}$.

Note that we consider only the case where the whole image is processed as one tile. For multiple tiles, data fidelity for each tile can be obtained independently as for separate images, while the TGV functional will be evaluated globally over all tiles. We further assume the same resolution level R for each color component. A generalization of our model to a color dependent resolution level $R(c)$, as it is possible within the JPEG 2000 standard, is straightforward but will be omitted for the sake of simplicity.

The component-wise wavelet transform operator of order R, denoted by $W = (W^1, W^2, W^3) : U \to U$, is now defined by repeatedly filtering each color component with the finite filters $(h_n)_n$ or $(g_n)_n$, followed by subsampling. The filters $(h_n)_n$ depend on the type of wavelet basis and correspond to the scaling functions $(\Phi_{j,k})_{j,k}$ as described in [8, Subsection 4.3.1], while the filters $(g_n)_n$ are defined by

$$g_n = (-1)^n h_{1-n}$$

and correspond to the wavelets $(\Psi_{j,k})_{j,k}$. Filtering both horizontally and vertically with $(h_n)_n$, for instance, decreases the resolution level by one step while involving the filters $(g_n)_n$ results in horizontal, vertical or diagonal detail information. Note that we use symmetric boundary extension for the filtering operations. In total, the filtering operations map the inner products $(u^c, \Phi_{0,k})$, describing an image $u \in U$, to inner products as in (29) that can be compared with the given JPEG 2000 data.

Using the dual filters $(\tilde{h}_n)_n$ and $(\tilde{g}_n)_n$, which correspond to the scaling function and mother wavelet, respectively, of the dual wavelet basis as described in [8, Subsection 4.3.1], the inverse wavelet transform operator

$$W^{-1} = ((W^1)^{-1}, (W^2)^{-1}, (W^3)^{-1})$$

can be defined component-wise by upsampling and filtering with the dual filters, again using symmetric boundary extension.

Suitably summing up all data intervals as defined in (28) to

$$(J^c_{i,j})_{i,j,c}, \quad \text{with } 0 \leq i,j < N,\, c \in \{1,2,3\},$$

the discrete data set for the JPEG 2000 decompression process can then be given as

$$U_D = \{u \in U \,|\, (Wu)^c_{i,j} \in J^c_{i,j},\, 0 \leq i,j < N,\, c \in \{1,2,3\}\}. \tag{30}$$

As in Subsection 3.1 we consider a discrete version of the TGV functional of order two defined on U as in Equation (3). Again, with these prerequisites, the discrete minimization problem for artifact free JPEG 2000 decompression can be written as

$$\min_{u \in U} \mathrm{TGV}^2_\alpha(u) + \mathcal{I}_{U_D}(u). \tag{31}$$

Now as the wavelet transformation operator W is not orthogonal in general, we cannot use the same solution strategy as for the JPEG case, since a closed form of the projection to the data set is no longer available. Hence we use a slight reformulation of the minimization problem which will result in the introduction of an additional dual variable for the data constraint. For that purpose, first note that, with $D = \{w \in U | w^c_{i,j} \in J^c_{i,j}\}$, the convex indicator function for the data set U_D can be written as $\mathcal{I}_{U_D}(u) = \mathcal{I}_D(Wu)$. Defining the spaces $X := U \times V$, $Y := V \times W$ and $Z = Y \times U$, and using again the function $F : Y \to \mathbb{R}$ and the operator $K : X \to Y$ as in (14) and (15), we define $\mathbf{F} : Z \to \mathbb{R}$ and $\mathbf{K} : X \to Z$ by

$$\mathbf{F}(z) = \mathbf{F}(x, w) = F(x) + \mathcal{I}_D(w)$$

and

$$\mathbf{K} = \begin{bmatrix} \nabla & -I \\ 0 & \mathcal{E} \\ W & 0 \end{bmatrix}.$$

Then, the optimization problem for JPEG 2000 decompression (31) can be posed as

$$\min_{x \in X} \mathbf{F}(\mathbf{K}x). \tag{32}$$

Existence of a solution to this problem as well as equivalence to a dual and saddlepoint problem are summarized in the following proposition:

Proposition 4.1. *There exists a solution $\hat{u}$ to* (31) *and $\hat{u}$ is optimal if and only if there exists $\hat{v}$ such that $\hat{x} = (\hat{u}, \hat{v})$ solves* (32). *Further, there exists a solution to the dual problem, given by*

$$\max_{z \in Z} -\mathcal{I}_{\{0\}}(-\mathbf{K}^* z) - \mathbf{F}^*(z), \tag{33}$$

and the saddle point problem

$$\min_{x \in X} \max_{z \in Z} (\mathbf{K}x, z) - \mathbf{F}^*(z), \tag{34}$$

and $\hat{x}, \hat{z}$ are solutions to the primal problem (32) *and the dual problem* (33), *respectively, if and only if $(\hat{x}, \hat{z})$ solves the saddle point problem* (34).

Proof. Existence of a solution to problem (31) follows by similar argumentation as in the infinite dimensional setting (see [8]). Equivalence of (31) to the primal problem (32) is obvious. Hence, following [18, Proposition III.3.1] it suffices to show existence of a solution to the dual problem and equality of the primal and dual problem at optimal points, to conclude all claims. Using [3, Corollary 2.3] this follows provided that

$$\bigcup_{\lambda \geq 0} \lambda[\mathrm{dom}(\mathbf{F}) - \mathbf{K}(\mathrm{dom}(\mathbf{0}))] = Z,$$

where $\mathbf{0} : X \to 0$ is the zero mapping. But this is satisfied since, given any $w_0 \in U_D$ fixed, any (p, q, w) can be written as

$$\begin{pmatrix} p \\ q \\ w \end{pmatrix} = \begin{pmatrix} p + \nabla W^{-1}(w_0 - w) \\ q \\ w_0 \end{pmatrix} - \mathbf{K} \begin{pmatrix} W^{-1}(w_0 - w) \\ 0 \end{pmatrix} \in \mathrm{dom}(\mathbf{F}) - \mathbf{K}(\mathrm{dom}(\mathbf{0})). \qquad \square$$

Again, optimality conditions for the discrete saddle-point problem will be helpful for a stopping criterion later on and follow by standard arguments (see [8, Theorem 3.3]).

Proposition 4.2. *Let the assumptions of Proposition 4.1 be satisfied. Then, there exists a solution to* (34) *and* $(\hat{x},\hat{z}) = (\hat{u},\hat{v},\hat{p},\hat{q},\hat{w})$ *solves* (34) *if and only if*

- $\begin{pmatrix} -\operatorname{div}\hat{p} + W^*\hat{w} \\ -\hat{p} - \operatorname{div}\hat{q}\end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$,
- *For all* $0 \le i < N$, $0 \le j < N$,
$$\begin{cases} \hat{p}_{i,j} = \alpha_1 \dfrac{(\nabla\hat{u}-\hat{v})_{i,j}}{|(\nabla\hat{u}-\hat{v})_{i,j}|_V} \text{ if } (\nabla\hat{u}-\hat{v})_{i,j} \neq 0 \text{ and } |\hat{p}_{i,j}|_V \le \alpha_0 \text{ else,} \\ \hat{q}_{i,j} = \alpha_0 \dfrac{(\mathcal{E}\hat{v})_{i,j}}{|(\mathcal{E}\hat{v})_{i,j}|_W} \text{ if } (\mathcal{E}\hat{v})_{i,j} \neq 0 \text{ and } |\hat{q}_{i,j}|_W \le \alpha_1 \text{ else,} \end{cases}$$
- $\hat{u} \in U_D$ *and*
$$\begin{cases} \hat{w}^c_{i,j} \ge 0 & \text{if } (W\hat{u})^c_{i,j} = \sup(J^c_{i,j}) \neq \inf(J^c_{i,j}), \\ \hat{w}^c_{i,j} \le 0 & \text{if } (W\hat{u})^c_{i,j} = \inf(J^c_{i,j}) \neq \sup(J^c_{i,j}), \\ \hat{w}^c_{i,j} = 0 & \text{if } (W\hat{u})^c_{i,j} \in \operatorname{int}(J^c_{i,j}). \end{cases}$$

4.2 Practical implementation

To solve (34) numerically, we again employ the abstract primal-dual algorithm given in Subsection 2.2. For that purpose, we need an explicit form of W^*, the adjoint of W, the proximal mapping $(I+\sigma\partial\mathbf{F}^*)^{-1}$ and an estimate on $\|\mathbf{K}\|$.

The adjoint of W**:** Since the filters $(h_n)_n$, $(g_n)_n$ for the forward wavelet transform are symmetric, and also the boundary extension in the evaluation of W is symmetric, W^* can be obtained by applying W^{-1} with $(h_n)_n$, $(g_n)_n$ instead of $(\tilde{h}_n)_n$, $(\tilde{g}_n)_n$.

The proximal mapping $(I+\sigma\partial\mathbf{F}^*)^{-1}$**:** First note that $\mathbf{F}^*$ can be decomposed as $\mathbf{F}^*(y^*,w^*) = F^*(y^*) + I^*_D(w^*)$ and hence $(I+\sigma\partial\mathbf{F}^*)^{-1}$ can be written as

$$(I+\sigma\partial\mathbf{F}^*)^{-1}(y,w) = \begin{pmatrix} (I+\sigma\partial F^*)^{-1}(y) \\ (I+\sigma\partial\mathcal{I}^*_D)^{-1}(w)\end{pmatrix},$$

where, as already discussed in Subsection 3.1, $(I+\sigma\partial F^*)^{-1}(y)$ is given as

$$(I+\sigma\partial F^*)^{-1}(y) = (I+\sigma\partial F^*)^{-1}(p,q) = \big(\operatorname{proj}_{\alpha_1}(p), \operatorname{proj}_{\alpha_0}(q)\big) \tag{35}$$

with $\operatorname{proj}_{\alpha_1}, \operatorname{proj}_{\alpha_0}$ as in Equation (19). As $\mathcal{I}^*_D$ can also be decomposed to

$$\mathcal{I}^*_D(w^*) = \sum_{i,j,c} \mathcal{I}^*_{J^c_{i,j}}((w^*)^c_{i,j}),$$

the resolvent $(I+\sigma\partial\mathcal{I}^*_D)^{-1}(w)$ can be obtained component-wise: In case $J=[l,o]$ is a bounded interval, a calculation already done in the proof of Proposition 3.3 again yields

$$\mathcal{I}^*_J(r^*) = \sup_{r\in J}(r,r^*) = \frac{l+o}{2}r^* + |\frac{o-l}{2}r^*|. \tag{36}$$

From this, one can deduce that $r^* = (I+\sigma\partial\mathcal{I}^*_J)^{-1}(r)$ is equivalent to a shrinkage operation according to

$$\begin{cases} r^* = r - \sigma\frac{l+o}{2} - \frac{o-l}{2}\sigma & \text{if } r - \sigma\frac{l+o}{2} > \frac{o-l}{2}\sigma, \\ r^* = r - \sigma\frac{l+o}{2} + \frac{o-l}{2}\sigma & \text{if } r - \sigma\frac{l+o}{2} < -\frac{o-l}{2}\sigma, \\ r^* = 0 & \text{if } |r - \sigma\frac{l+o}{2}| \le \frac{o-l}{2}\sigma. \end{cases} \tag{37}$$

In the case $J = \mathbb{R}$, it is immediate that

$$\mathcal{I}_J^*(r^*) = \mathcal{I}_{\{0\}}(r^*) \quad \text{and} \quad (I + \sigma\partial\mathcal{I}_J^*)^{-1}(r) = 0.$$

Hence, denoting by $\mathcal{N}$ the set of all indices (i, j, c) such that $J_{i,j}^c$ is bounded, i.e, is of the form $J_{i,j}^c = [l_{i,j}^c, o_{i,j}^c]$ with $l_{i,j}^c, o_{i,j}^c \in \mathbb{R}$, we get that

$$(I + \sigma\partial\mathbf{F}^*)^{-1}(x, w) = \begin{pmatrix} \operatorname{proj}_{\alpha_1}(v) \\ \operatorname{proj}_{\alpha_2}(w) \\ \operatorname{shrink}_{J,\mathcal{N}}(w, \sigma) \end{pmatrix},$$

with the shrinkage operator $\operatorname{shrink}_{J,\mathcal{N}}$ defined as

$$(\operatorname{shrink}_{J,\mathcal{N}}(w, \sigma))_{i,j}^c = \begin{cases} \begin{cases} w_{i,j}^c - \sigma o_{i,j}^c & \text{if } w_{i,j}^c > \sigma o_{i,j}^c, \\ w_{i,j}^c - \sigma l_{i,j}^c & \text{if } w_{i,j}^c < \sigma l_{i,j}^c, \\ 0 & \text{if } w_{i,j}^c \in [\sigma l_{i,j}^c, \sigma o_{i,j}^c], \end{cases} & \text{if } (c, i, j) \in \mathcal{N}, \\ 0 & \text{else.} \end{cases} \tag{38}$$

The norm of K: An estimate on $\|\mathbf{K}\|$ needs an estimate on $\|W\|$. Even though such an estimate can be obtained analytically, $\|W\|$ depends on the level of wavelet decomposition and can get quite high. For example, using a decomposition level of 5 and the CDF 9/7 filters, we can estimate $\|K\| \leq 2839.97$ and, choosing equal stepsizes, consequently get

$$\sigma = \tau \approx 0.000352.$$

Compared to the JPEG case, where $\sigma = \tau \approx 0.2965$, this is clearly too small for efficient numerical solution. Even though our estimate on $\|W\|$ certainly was not optimal, one can get lower bounds on $\|W\|$ in particular situations which still yield impracticable small stepsizes. To overcome this difficulty, we propose to choose the stepsize adaptively as explained in Subsection 2.2 and defined in (9). In practice, the adaptive stepsize choice reduces the stepsize only a few times during the first iterations and allows stepsizes in the range of 0.3, and consequently, significantly accelerates convergence.

A scheme of implementation for the JPEG 2000 decompression process can then be given in Algorithm 3. Note that there, J denotes the collection of all data intervals and d the matrix where each entry is the midpoint of the corresponding interval (0 if the interval is $\mathbb{R}$). Further $\nabla, \mathcal{E}, \operatorname{div}, \operatorname{div}$ are again the component-wise gradient and divergence operators as defined in Subsection 3.1.

Stopping rule

In order to validate our numerical solution, we again seek for a suitable stopping rule that allows to estimate optimality in terms of the objective function. However, as, in contrast to the JPEG decompression setting, we cannot ensure the current iterates being contained in the data set U_D, we cannot expect to get an estimate on

$$0 \leq \mathbf{F}(\mathbf{K}x_n) - \mathbf{F}(\mathbf{K}\hat{x})$$

as $\mathbf{F}(\mathbf{K}x_n)$ might be infinity. Our solution is to relax the data constraint as follows. Given some constants $(C_{i,j}^c)_{(i,j,c)\in\mathcal{N}}$ with $\mathcal{N}$ the set of indices where the intervals are bounded, i.e., $J_{i,j}^c = [l_{i,j}^c, o_{i,j}^c]$ for $(i, j, c) \in \mathcal{N}$, we define

$$I_{D_C}(w) = \sum_{w_{i,j}^c \notin J_{i,j}^c} C_{i,j}^c \max\{w_{i,j}^c - o_{i,j}^c, l_{i,j}^c - w_{i,j}^c\}.$$

Algorithm 3 Scheme of implementation for JPEG 2000 decompression

```
function TGV-JP2($J_{\mathrm{comp}}$)
    $(d, J) \leftarrow$ Decoding of JPEG 2000 object $J_{\mathrm{comp}}$
    $\mathcal{N} \leftarrow J$
    $u \leftarrow W^{-1}(d)$
    $v \leftarrow 0, \overline{u} \leftarrow u, \overline{v} \leftarrow 0, p \leftarrow 0, q \leftarrow 0, w \leftarrow 0$
    choose $\sigma, \tau > 0$
    repeat
        $p \leftarrow \mathrm{proj}_{\alpha_1}(p + \sigma(\nabla \overline{u} - \overline{v}))$
        $q \leftarrow \mathrm{proj}_{\alpha_0}(q + \sigma \mathcal{E} \overline{v})$
        $w \leftarrow \mathrm{shrink}_{J,N}(w + \sigma W \overline{u}, \sigma)$
        $u_+ \leftarrow u - \tau(-\operatorname{div} p + W^* w)$
        $v_+ \leftarrow v - \tau(-p - \operatorname{div} q)$
        $\overline{u} \leftarrow (2u_+ - u),\ \overline{v} \leftarrow (2v_+ - v)$
        $\sigma_+ \leftarrow \mathcal{S}(\sigma\tau, \frac{\|(u_+,v_+)-(u,v)\|_X}{\|\mathbf{K}((u_+,v_+)-(u,v))\|_Z}),\ \tau_+ \leftarrow \sigma_+$
        $u \leftarrow u_+,\ v \leftarrow v_+$
    until Stopping criterion fulfilled
    return $u_+$
end function
```

Note that I_{D_C} is a relaxed version of $\mathcal{I}_D$ as coefficients outside of the data intervals are penalized with $C^c_{i,j}$ instead of infinity. An easy calculation shows that

$$I^*_{D_C}(w^*) = \sum_{(i,j,c)\in\mathcal{N}} \mathcal{I}_{\{|\cdot| \leq C^c_{i,j}\}}((w^*)^c_{i,j}) + \frac{l^c_{i,j} + o^c_{i,j}}{2} w^c_{i,j} + |\frac{o^c_{i,j} - l^c_{i,j}}{2} w^c_{i,j}| + \sum_{(i,j,c)\notin\mathcal{N}} \mathcal{I}_{\{0\}}(w^c_{i,j}),$$

in particular $I^*_{D_C}(w^*) = \mathcal{I}^*_D(w^*)$ if $|w^c_{i,j}| \leq C^c_{i,j}$ for all $(i,j,c) \in \mathcal{N}$. Now given $\hat{x} = (\hat{u}, \hat{v})$ and $\hat{z} = (\hat{p}, \hat{q}, \hat{w})$ an optimal solution to the saddle point problem (34) and choosing $\hat{C}^c_{i,j} = |\hat{w}^c_{i,j}|$ for all $(i,j,c) \in \mathcal{N}$, we can estimate for any $x = (u,v) \in X$

$$\begin{aligned}
F(Kx) + I_{D_{\hat{C}}}(Wu) &= \sup_{s=(p,q,w)} (\mathbf{K}x, s)_Z - F^*(p,q) - I^*_{D_{\hat{C}}}(w) \\
&\geq (\mathbf{K}x, \hat{z})_Z - F^*(\hat{p}, \hat{q}) - I^*_{D_{\hat{C}}}(\hat{w}) \\
&\geq (\mathbf{K}\hat{x}, \hat{z})_Z - F^*(\hat{p}, \hat{q}) - \mathcal{I}^*_D(\hat{w}) \\
&= \sup_{s=(p,q,w)} (\mathbf{K}\hat{x}, s)_Z - F^*(p,q) - \mathcal{I}^*_D(w) \\
&= \mathbf{F}(\mathbf{K}\hat{x})
\end{aligned}$$

from which we can proceed as in the JPEG setting to get, for any $z \in Z$,

$$\begin{aligned}
0 \leq F(Kx) + I_{D_{\hat{C}}}(Wu) - \mathbf{F}(\mathbf{K}\hat{x}) &= F(Kx) + I_{D_{\hat{C}}}(Wu) - \left(\sup_{s\in Z} (\mathbf{K}\hat{x}, s)_Z - \mathbf{F}^*(s) \right) \\
&\leq F(Kx) + I_{D_{\hat{C}}}(Wu) - (\mathbf{K}\hat{x}, z)_Z + \mathbf{F}^*(z).
\end{aligned} \tag{39}$$

This allows to define a suitable stopping rule as follows.

Proposition 4.3. *Let $\gamma > 1$, $1 \leq r \leq \infty$ $x_n = (u_n, v_n), z_n = (p_n, q_n, w_n)$ be the iterates of Algorithm 3 and $(\hat{u}, \hat{v}), (\hat{p}, \hat{q}, \hat{w})$ be an optimal solution to (34). Then, defining*

$$\begin{aligned}\mathcal{G}(x_n, z_n) := \quad & F(Kx_n) + I_{D_{C_n}}(Wu_n) + T_n \| \operatorname{div}^2 \tilde{q}_n - W^* w_n \|_{r'} \\ & + \sum_{(i,j,c) \in \mathcal{N}} \frac{l^c_{i,j} + o^c_{i,j}}{2} (w_n)^c_{i,j} + |\frac{o^c_{i,j} - l^c_{i,j}}{2} (w_n)^c_{i,j}|\end{aligned} \tag{40}$$

with $(C_n)^c_{i,j} = \gamma |(w_n)^c_{i,j}|$, $T_n := \gamma \|u_n\|_r$ and $\tilde{q}_n = \beta_n q_n$ with $\beta_n := \frac{\alpha_1}{\max(\alpha_1, \|\operatorname{div} q_n\|_\infty)}$, we get that

$$\mathcal{G}(x_n, z_n) \to 0 \text{ as } n \to \infty$$

and, additionally,

$$0 \leq F(Kx_n) + I_{D_{\hat{C}}}(Wu_n) - \mathrm{TGV}^2_\alpha(\hat{u}) \leq \mathcal{G}(x_n, z_n) \tag{41}$$

whenever $T_n \geq \|\hat{u}\|_r$ and $(C_n)^c_{i,j} \geq |\hat{w}^c_{i,j}|$ for all $0 \leq i < N$, $0 \leq j < N$, $c \in \{1, 2, 3\}$, which is eventually the case.

Proof. The claimed convergence to zero follows, with similar techniques as employed in the proof of Proposition 3.3, from convergence of the iterates, the optimality conditions given in Proposition 4.2 and strict duality as shown in the proof of Proposition 4.1. Now inserting $x_n = (u_n, v_n)$, $\tilde{z}_n = (-\operatorname{div} \tilde{q}_n, \tilde{q}_n, w_n)$ in (39) the last term reduces to

$$F(Kx) + I_{D_{\hat{C}}}(Wu) - (\mathbf{K}\hat{x}, z)_Z + \mathbf{F}^*(z) = F(Kx_n) + I_{D_{\hat{C}}}(Wu_n) - (\hat{u}, \operatorname{div}^2 \tilde{q}_n + Ww_n)_U + \sup_{s \in D} (s, w_n)_U. \tag{42}$$

As, due to the shrinkage operator in Algorithm 3, all indices of w_n corresponding to unbounded intervals are zero, the supremum in the above expression can be written explicitly as

$$\sup_{s \in D} (s, w_n)_U = \sum_{i,j,c} \frac{l^c_{i,j} + o^c_{i,j}}{2} (w_n)^c_{i,j} + |\frac{o^c_{i,j} - l^c_{i,j}}{2} (w_n)^c_{i,j}|.$$

Combining this with (42) and the estimate (39) and assuming that $T_n \geq \|\hat{u}\|_r$ and $(C_n)^c_{i,j} \geq |\hat{w}^c_{i,j}|$ the estimate (41) follows. Finally, due to convergence of the iterates and $\gamma > 1$, both $T_n \geq \|\hat{u}\|_r$ and $(C_n)^c_{i,j} \geq |\hat{w}^c_{i,j}|$ are satisfied after finitely many iterations. □

This allows, for given $\epsilon > 0$, to use $\mathcal{G}(x_n, y_n) < \epsilon$ as stopping criterion and provides, at least in the limit, a suitable estimate of the error in terms of functional values. Note that we cannot expect to get the estimate

$$\mathcal{G}(x_n, z_n) \geq F(Kx_n) - \mathrm{TGV}^2_\alpha(\hat{u}) \geq 0$$

since, as already mentioned, the iterates $(u_n)_n$ are only contained in the data set U_D in the limit and thus it is possible that $F(Kx_n) < \mathrm{TGV}^2_\alpha(\hat{u})$. This was also observed in numerical experiments.

4.3 Numerical experiments

The aim of this subsection is to provide numerical results obtained with our framework for improved decompression of JPEG 2000 images. We used Algorithm 3 with adaptive stepsize choice, $\theta = 0.95$ and a stepsize initialization of $\sigma = \tau = 1/3$ for all experiments in this section.

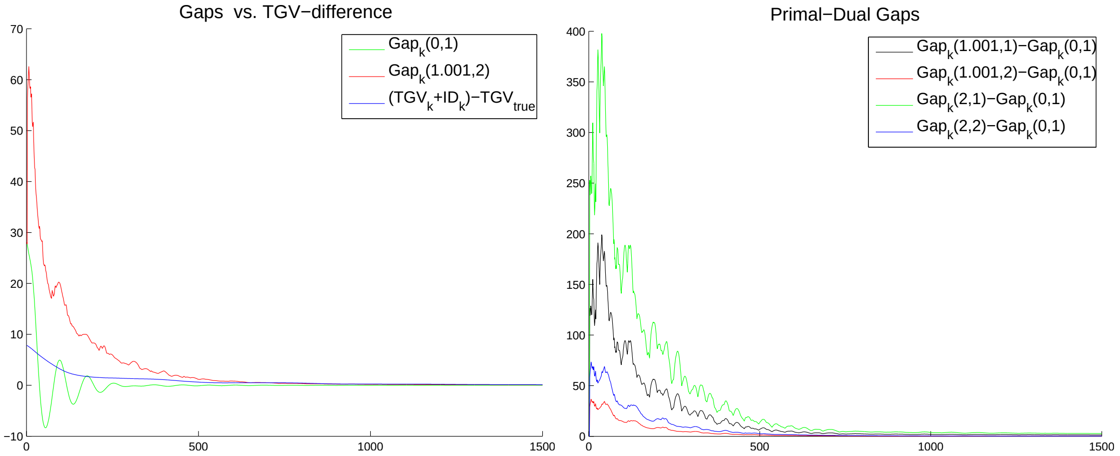

Figure 3: Left: Comparison of modified primal-dual gap for $\gamma = 1.001$, $r = 2$ and $\gamma = 0$ to difference of $\mathrm{TGV}(u_k) + \mathcal{I}_{D_{C_k}}(u_k)$ and optimal TGV value (obtained with adaptive stepsize and 15000 iterations). As one can see, the modified primal-dual gap with $\gamma = 0$ violates the estimate $\mathcal{G}(x_k, y_k) \geq \mathrm{TGV}(u_k) + I_{D_{C_k}}(Wu_k) - \mathrm{TGV}(\hat{u})$. Right: Difference of modified primal-dual gap for different choices of γ, r to modified primal-dual gap obtained with $\gamma = 0$.

As stopping rule, we use a normalization of the modified primal-dual gap $\mathcal{G}$ as in (40), defined by

$$\overline{\mathcal{G}}(x_n, y_n) = \frac{\mathcal{G}(x_n, y_n)}{N^2}, \tag{43}$$

with N^2 the number of image pixels. As for the application to JPEG decompression, the normalization is motivated by making $\mathcal{G}$ image size independent and getting an estimate on an average pixel error; remembering that, in the limit, we have the estimate

$$0 \leq F(x_n) + I_{D_{C_n}}(Wu_n) - \mathrm{TGV}^2_\alpha(\hat{u}) \leq \mathcal{G}(x_n, y_n).$$

In contrast to the JPEG decompression setting, we did not implement CPU or GPU optimized code for the TGV based JPEG 2000 decompression algorithm. Considering computation times, however, one could expect that each iteration step of an optimized JPEG 2000 decompression scheme would take about as long as for the JPEG decompression setting. Indeed, from the computational viewpoint, the only considerable difference is that instead of a Block-DCT transform, a wavelet transform is used.

Primal-dual gap parameters

As for the application to JPEG decompression, we again have free choice of the parameters γ and r for evaluation of the modified primal-dual gap: Figure 3 evaluates different choices of γ and r. As can be seen in the left plot the choice $\gamma = 0$ violates the estimate (41), but the choice $\gamma = 1.001$, $p = 2$, $r = 2$ does not. The observation that, in contrast to JPEG decompression, the choice $\gamma = 0$ is not feasible can be explained by the fact that this time the unbounded data part, and, consequently the choice of γ, has much more influence. As can be seen in Figure 3 on the right, even not violating estimate (41), the choices $\gamma = 1.001$ and $r = 2$ result again in the lowest modified primal-dual gap. Thus, we used this choice for all subsequent experiments.

Experimental results for decompression

We now consider numerical results obtained with the proposed method for different JPEG 2000 compressed images. If not mentioned otherwise, we always require the normalized modified primal-dual gap to be below 10^{-1} as stopping criterion and again fix the ratio between α_0 and α_1 for the evaluation of the TGV_α^2 functional as $\frac{\alpha_0}{\alpha_1} = \sqrt{2}$ based on empirical observations. Figure 4 compares the standard reconstruction with the TGV based reconstruction. As one can see, in the TGV based reconstruction, all wavelet artifacts have been removed while edges are kept sharp. In general the reconstruction quality is strongly improved and yields more natural and visually more appealing images. The two top rows of Figure 4 show a compression and reconstruction of the same parrot image, once without tiling and once using 4 tiles. Even at the same bit rate, the image with tiling looks worse as the tile boundaries are clearly visible. In the improved reconstruction, these artificial edges as well as the stronger wavelet artifacts have been removed completely. But also the improved reconstruction quality is slightly worse when using multiple tiles. The larger number of iterations (3975, 4580 and 5056 for the two parrots and the peppers image, respectively), compared to JPEG reconstruction, necessary to satisfy the stopping rule might be due to the unboundedness of some data intervals for JPEG 2000 decompression. However, as we will see in Figure 6 later on, this bound is merely important to ensure optimality, in practical applications a much lower iteration number is sufficient to obtain a reconstruction visually almost indistinguishable from the optimal one.

Figure 5 allows a more detailed comparison of the standard and TGV based reconstruction for a color test image. As one can see in the difference images below, the total energy of the error has been significantly reduced with the TGV based reconstruction and the error is of a less visible type.

Figure 6 now allows to compare the reconstruction quality for the parrot image by performing less iterations. It depicts the standard decompression and reconstructions obtained using 100 iterations, $\overline{\mathcal{G}}(x_n, y_n) < 1$ (836 iterations) and again $\overline{\mathcal{G}}(x_n, y_n) < 0.1$ (3975 iterations) as stopping rule. As one can see, even after 100 iterations the JPEG 2000 artifacts are already removed and the image is comparable to the optimal solution at 4111 iterations. However, in contrast to JPEG decompression, for such suboptimal reconstructions a perfect fit to data cannot be ensured and is not satisfied in practice.

5 Variational image zooming

5.1 Discrete framework

This section deals with a discrete version of the variational zooming approach as described in [8, Subsection 4.4]. For simplicity, we consider only grayscale images and define $U = \mathbb{R}^{N \times N}$, $N \in \mathbb{N}$, to be the space of discrete, high resolution images. Given a Riesz basis $(a_{i,j})_{i,j}$ of U, we define a subsampling operator $A : \mathbb{R}^{N \times N} \to \mathbb{R}^{M \times M}$ with $M < N$ as

$$(Au)_{i,j} = (a_{i,j}, u)_U \quad \text{for } 0 \le i, j < M.$$

Our goal is to perform a TGV regularized inversion of this type of subsampling, i.e., given low resolution image data $(d_{i,j})_{i,j}$, to find an image $u \in U$ minimizing the TGV_α^2 function as in (3) such that

$$(Au)_{i,j} = d_{i,j} \quad \text{for all } 0 \le i, j < M.$$

A particular case of such a setting, which has also been considered in [6], arises when choosing $(a_{i,j})_{i,j}$ to be a wavelet basis of U, which is, for a given resolution level R, again denoted by

$$(\Phi_{R,k})_k \cup (\Psi_{j,k})_{j,k}. \tag{44}$$

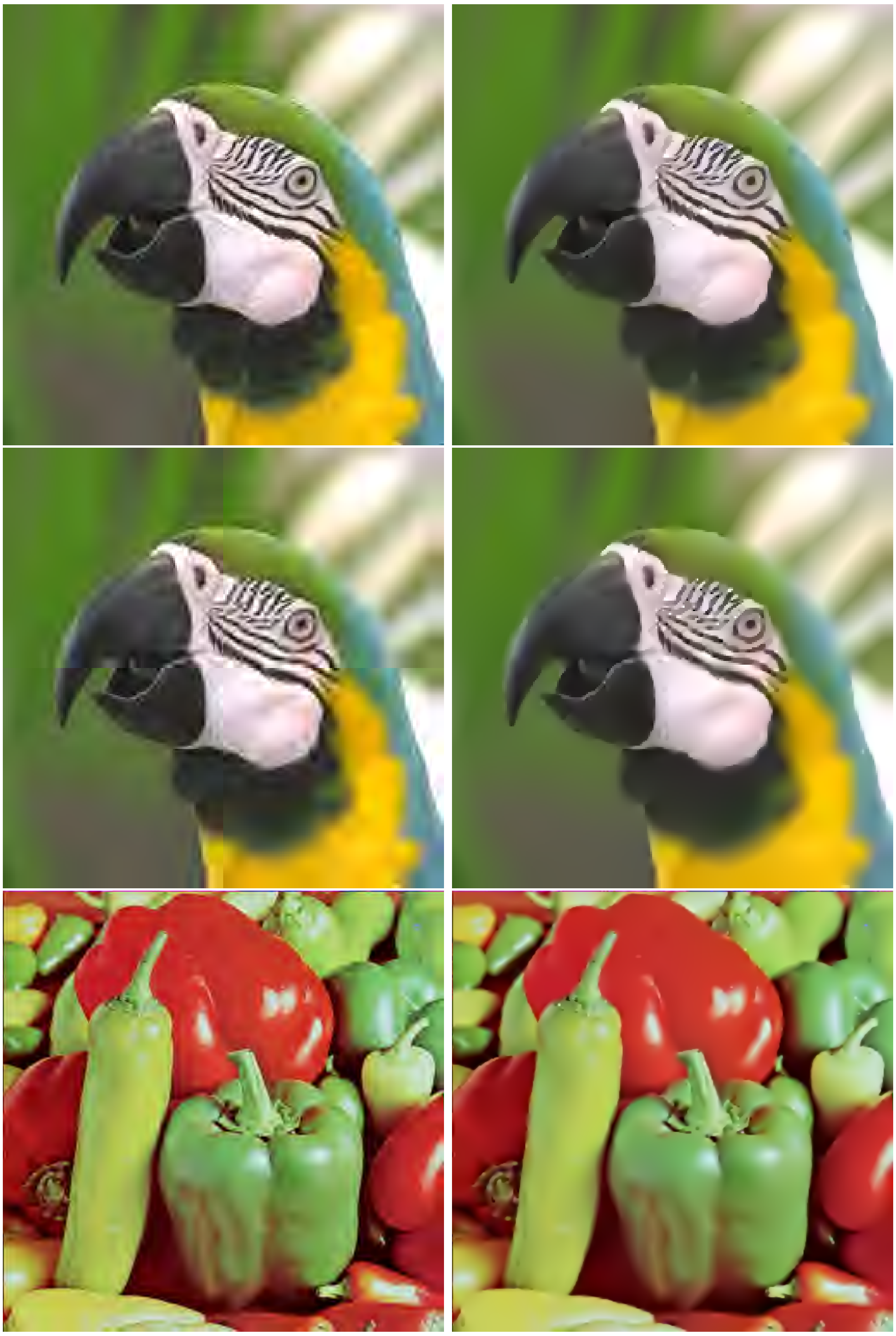

Figure 4: On the left: Standard decompression. On the right: TGV based reconstruction obtained with normalized modified primal-dual gap below 10^{-1} as stopping criterion. From top to bottom: Parrot image at 0.3 bpp (256×256 pixels, 3975 iterations), Parrot image with tiling at 0.3 bpp (256×256 pixels, 4580 iterations), Peppers image at 0.15 bpp (512×512 pixels, 5056 iterations).

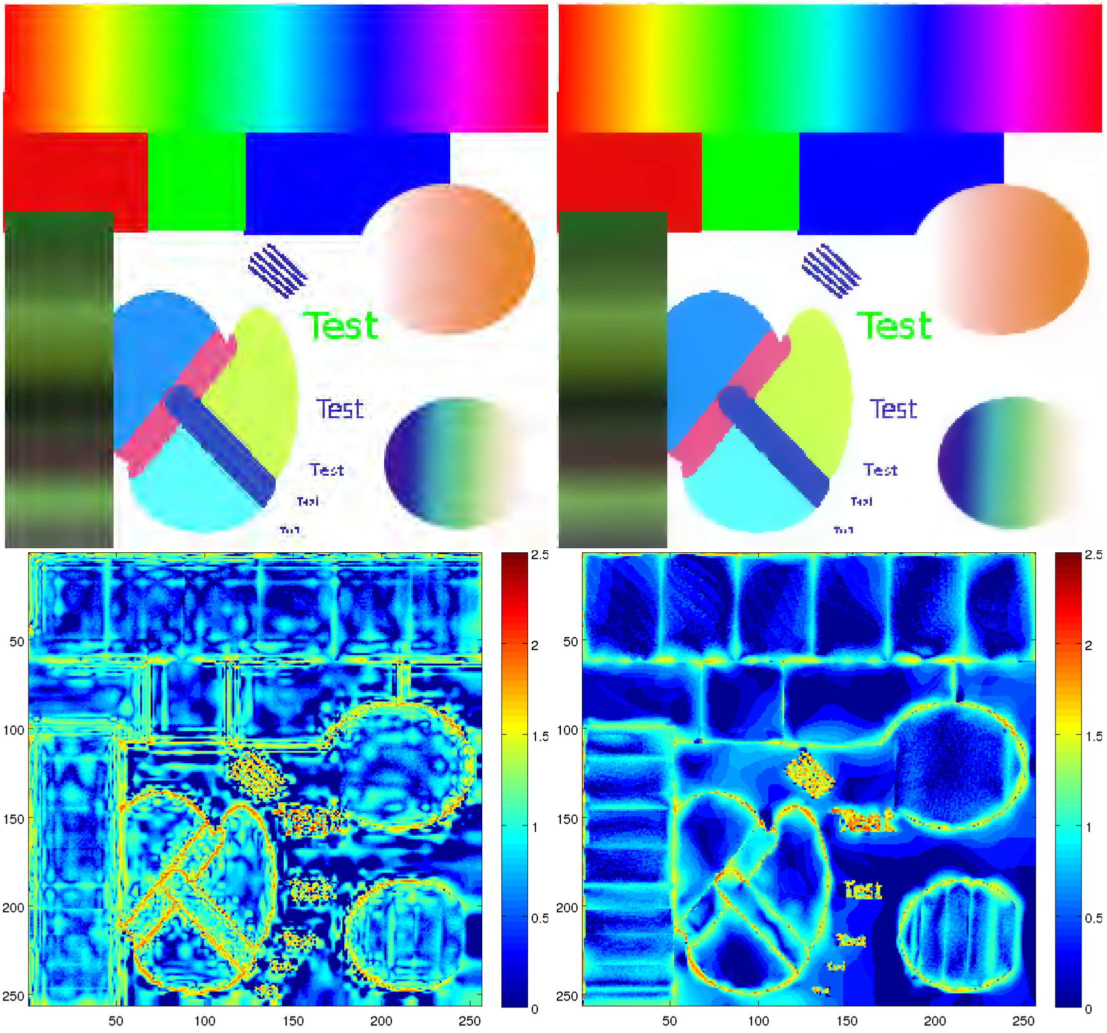

Figure 5: Left: Standard decompression, right: TGV based reconstruction. Top: Color test image at 0.6 bpp (256 × 256 pixels, 4144 iterations). Bottom: Visualization of the pointwise reconstruction error (logarithmic scale).

Then, possibly shifting resolution levels, we can choose $M = 2^{-R}N$ and define A to map high resolution images $u \in U$ to inner products with the scaling functions, i.e., $Au = ((u, \Phi_{R,k})_U)_k$ for $0 \leq k < M^2$. As these inner products correspond to a local averaging, we interpret each $(u, \Phi_{R,k})$ as pixel value and hence A indeed maps images $u \in U$ to low resolution images $Au \in \mathbb{R}^{2^{-R}N \times 2^{-R}N}$ (see also [8, Figure 5]). In particular, when choosing $(a_{i,j})_{i,j}$ to be a Haar wavelet basis, $(u, \Phi_{R,k})$ is a local average on a patch of size 2^R at position k.

Alternatively, one can also incorporate data from a given JPEG or JPEG 2000 compressed file to obtain a combined decompression and zooming approach. Choosing $(a_{i,j})_{i,j}$ for $0 \leq i, j < N$ to be either a block-wise cosine basis or a tile-wise wavelet basis with CDF 9/7 or Le Gall 5/3 wavelets, we can assume that a given file describes only a low resolution image $u \in \mathbb{R}^{M \times M}$ by providing interval bounds $J_{i,j}$ on the coefficients $(a_{i,j}, u)_U$ for $0 \leq i, j < M$ and reconstruct a

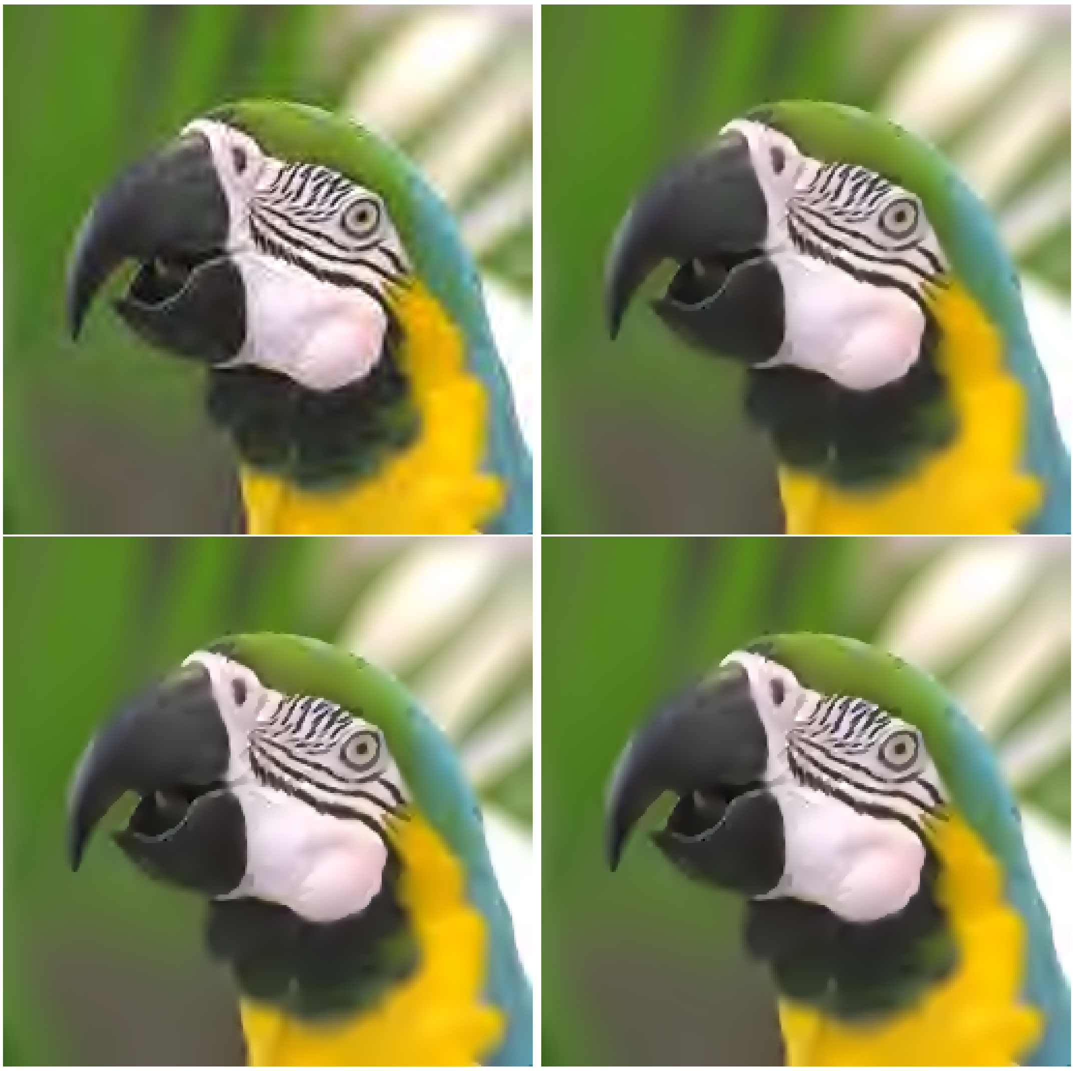

Figure 6: Decompression of parrot image (0.3 bpp, 256 × 256 pixels) using only a few number of iterations. Top left: Standard decompression, top right: TGV regularized decompression after 100 iterations, bottom left and right: TGV regularized decompression obtained with $\overline{\mathcal{G}}(x_n, y_n) < 1$ (836 iterations) and $\overline{\mathcal{G}}(x_n, y_n) < 1$ (3975 iterations) as stopping criterion, respectively.

high resolution image $u \in U$ such that $(Au)_{i,j} \in J_{i,j}$ for $0 \leq i, j < M$.

In both cases, the proposed zooming approach can be realized by solving

$$\min_{u \in U} \operatorname{TGV}_\alpha^2(u) + \mathcal{I}_{U_D}(u), \tag{45}$$

where

$$U_D = \{u \in U \,|\, (Au)_{i,j} \in J_{i,j} \text{ for } 0 \leq i, j < M\}, \tag{46}$$

with $J_{i,j}$ either being point intervals or intervals that are given from a compressed JPEG or JPEG 2000 file.

Now since the setting we used for the discrete minimization problem for JPEG 2000 decompression does not exclude the case of point intervals, all results, such as existence of a solution and the optimality condition, apply. Consequently, also the saddle point problem

$$\min_{x \in X} \max_{z \in Z} \, (\mathbf{K}x, z) - \mathbf{F}^*(z), \tag{47}$$

with $X, Z, \mathbf{K}, \mathbf{F}$ defined as in Subsection 4.1, where U_D is now given as in (46), is equivalent to (45) and can be solved by the primal-dual Algorithm 3. Also, the considerations concerning the norm of $\mathbf{K}$ and, consequently, the stepsizes σ, τ as well the estimates on the modified primal-dual gap $\mathcal{G}$ defined in (40) directly apply to the variational zooming model. In applications where the basis $(a_{i,j})_{i,j}$ is orthogonal, such as when $(a_{i,j})_{i,j}$ is a block cosine basis or a Haar wavelet basis, also the simpler algorithmic framework of JPEG compression as in Section 3.1 applies. The modified primal-dual gap given there, however, has to be modified accordingly to account for unbounded intervals.

5.2 Numerical experiments

This section presents numerical results for variational image zooming and combined decompression and zooming. As stopping rule for our experiments, we require the normalized modified primal-dual gap as in (43) to be below 10^{-1} for all experiments. Motivated by the discussion in Subsection 4.3 we again use adaptive stepsizes initialized with $\sigma = \tau = 1/3$ and $\gamma = 1.001$, $p = 2$ and $r = 2$ for the evaluation of the modified primal-dual gap. However, in contrast to the decompression setting, we now fix the ratio between α_0 and α_1 for evaluation of the TGV functional to 4 rather than $\sqrt{2}$ since, as we experienced, this choice improves reconstruction quality in zooming applications.

Figure 7 shows results for the fourfold magnification of a patch of the Barbara image containing a stripe structure, and a synthetic test image. For better comparability, we used the original image rather than a down-sampled version. Thus, the down-sampling procedure is not known and cannot favor any particular method, but also no original data is available. As one can see for the Barbara image, a Lanczos 2 filter [17] based zooming leads to blurring of the stripes while our method yields a reconstruction appearing much sharper. Using the CDF 9/7 wavelets results in the best reconstruction quality. In particular, we observe that not only the edges are preserved, but also the geometrical information is extended in a natural manner for the CDF 9/7 wavelet (as opposed to the Haar wavelet, where "geometrical staircasing" occurs). For the test image at the bottom of Figure 7, we can observe that, as one would expect, the Haar wavelet based approach keeps the block structure of the edges of the objects, but leads to sharp object boundaries and naturally smoothes homogeneous regions.

Figure 8 then shows results for the combined decompression and eightfold magnification of JPEG and JPEG 2000 decompressed images. As one can see, even in the difficult situation of having only highly subsampled and compressed data available, our method performs reasonably well. While with standard box upsampling, the artifacts resulting from compression are magnified, the do not appear in the TGV regularized upsampled images. Also, sharp boundaries of the hand are recovered reasonably well. Fine texture is, of course, lost due to high subsampling.

6 Conclusions

We have presented different applications of a general variational model for TGV regularized image reconstruction. After the model has been analyzed in function space setting in a previous work [8], we have focused on the discrete setting and have obtained a practical realization for JPEG

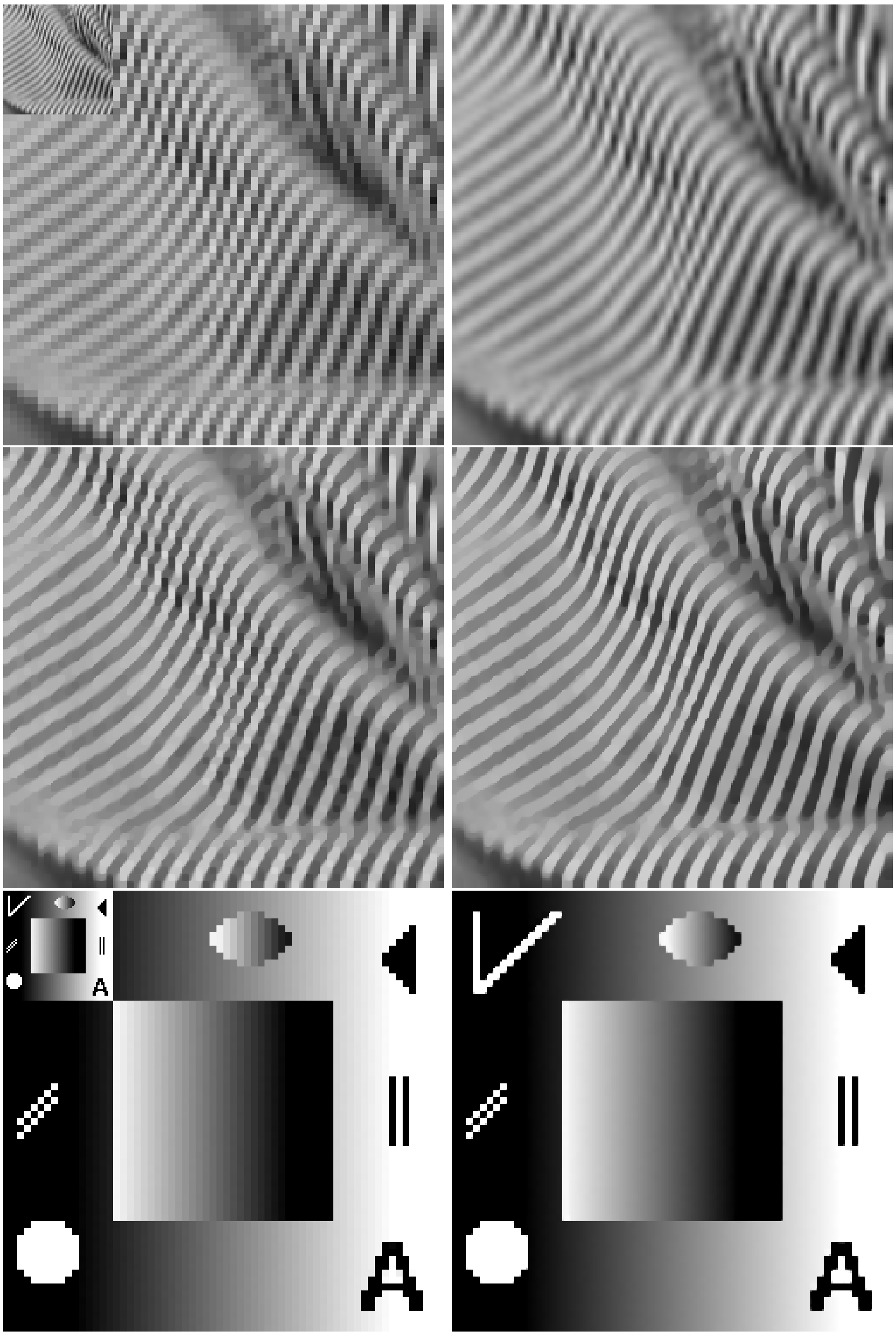

Figure 7: Top left: Low resolution and box-upsampled patch of Barbara images. Top right: 4 times magnifications by linear filtering with Lanczos 2 filter. Center left and right: 4 times magnification by TGV based variational zooming using the Haar (1948 iterations) and CDF 9/7 wavelet (1773 iterations) to model subsampling, respectively. Bottom: 4 times magnification with box-upsampling (left) and TGV based variational zooming using the Haar wavelet (right, 4680 iterations) of a test image.

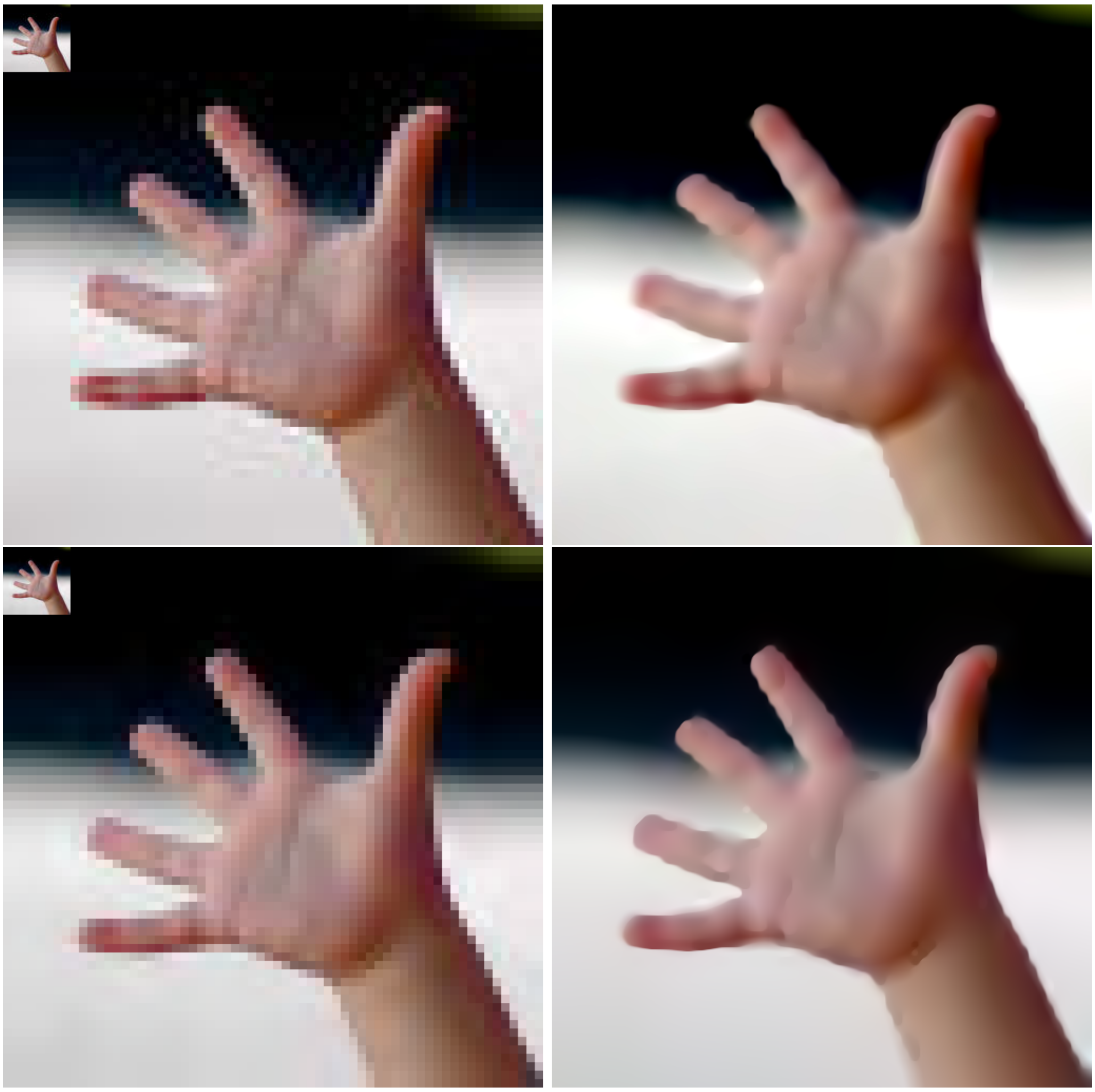

Figure 8: Upsampling by pixel repetition (left) and $\mathrm{TGV}_{\alpha}^{2}$ regularized, combined decompression and zooming (right) for a low resolution, JPEG (top, 3276 iterations) and JPEG 2000 (bottom, 11893 iterations) compressed image (both compressed to 2.96 bpp). Image by [21], licensed under CC-BY-2.0 (http://creativecommons.org/licenses/by/2.0/).

and JPEG 2000 decompression and a variational zooming method. In particular, a uniform algorithmic framework allowing to obtain a globally convergent algorithm for all applications has been presented. To ensure a reasonably fast convergence and optimality of the obtained solutions, we have proposed an adaptive stepsize strategy and a duality based stopping criterion. For JPEG decompression, we have also developed a multi-core CPU and GPU implementation that shows our approach to be suited for practical application. Numerical experiments further allow to compare results obtained with TV and second and third order TGV regularization and confirm a high visual quality of our obtained reconstruction.

References

[1] F. Alter, S. Durand, and J. Froment. Adapted total variation for artifact free decompression of JPEG images. *Journal of Mathematical Imaging and Vision*, 23:199–211, 2005.

[2] H. A. Aly and E. Dubois. Image up-sampling using total-variation regularization with a new observation model. *IEEE Transactions on Image Processing*, 14(10):1647–1659, 2005.

[3] H. Attouch and H. Brezis. Duality for the sum of convex functions in general Banach spaces. *Aspects of Mathematics and its Applications*, 34:125–133, 1986.

[4] K. Bredies and M. Holler. A total variation–based JPEG decompression model. *SIAM Journal on Imaging Sciences*, 5(1):366–393, 2012.

[5] K. Bredies and M. Holler. Artifact-free decompression and zooming of JPEG compressed images with total generalized variation. In *Computer Vision, Imaging and Computer Graphics. Theory and Application*, volume 359, pages 242–258. Springer, 2013.

[6] K. Bredies and M. Holler. A TGV regularized wavelet based zooming model. In *Scale Space and Variational Methods in Computer Vision*, volume 7893, pages 149–160. Springer, 2013.

[7] K. Bredies and M. Holler. Regularization of linear inverse problems with total generalized variation. *Journal of Inverse and Ill-posed Problems*, 22(6):871–913, 2014.

[8] K. Bredies and M. Holler. A TGV-based framework for variational image decompression, zooming and reconstruction. Part I: Analytics. *Submitted*, 2015.

[9] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.

[10] Kristian Bredies. Recovering piecewise smooth multichannel images by minimization of convex functionals with total generalized variation penalty. In *Efficient Algorithms for Global Optimization Methods in Computer Vision*, volume 8293 of *Lecture Notes in Computer Science*, pages 44–77. Springer Berlin Heidelberg, 2014.

[11] W. K. Carey, D. B. Chuang, and S. S. Hemami. Regularity-preserving image interpolation. *IEEE Transactions on Image Processing*, 8(9):1293–1297, Sep 1999.

[12] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:88–97, 2004.

[13] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2011.

[14] R. H. Chan, J. Yang, and X. Yuan. Alternating direction method for image inpainting in wavelet domains. *SIAM Journal on Imaging Sciences*, 4(3):807–826, 2010.

[15] T. F. Chan, J. Shen, and H.-M. Zhou. Total variation wavelet inpainting. *Journal of Mathematical Imaging and Vision*, 25(1):107–125, 2006.

[16] T. Chen, H. R. Wu, and B. Qiu. Image interpolation using across-scale pixel correlation. In *Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, volume 3, pages 1857–1860, 2001.

[17] C. E. Duchon. Lanczos Filtering in One and Two Dimensions. *Journal of Applied Meteorology and Climatology*, 18(8):1016–1022, 1979.

[18] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems.* SIAM, 1999.

[19] M. Frigo and S. G. Johnson. The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005. Special issue on "Program Generation, Optimization, and Platform Adaptation".

[20] N. Kaulgud and U. B. Desai. Image zooming: Use of wavelets. In *The International Series in Engineering and Computer Science*, volume 632, pages 21–44. Springer, 2002.

[21] J. Kubina. http://flickr.com/photos/kubina/42275122, 2008.

[22] F. Malgouyres and F. Guichard. Edge direction preserving image zooming: A mathematical and numerical analysis. *SIAM Journal on Numerical Analysis*, 39:1–37, 2001.

[23] NVIDIA. NVIDIA CUDA programming guide 2.0. NVIDIA Cooperation, 2008.

[24] R. Oktem. Regularization-based error concealment in JPEG 2000 coding scheme. *IEEE Signal Processing Letters*, 14(12):956–959, 2001.

[25] S. Ono and I. Yamada. Optimized JPEG image decompression with super-resolution interpolation using multi-order total variation. In *20th IEEE International Conference on Image Processing (ICIP)*, pages 474–478, Sept 2013.

[26] OpenMP Architecture Review Board. Openmp application program interface, version 3.1. http://www.openmp.org, 2011.

[27] G. K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, 1991.

[28] Y.-W. Wen, R. H. Chan, and A. M. Yip. A primal-dual method for total variaton-based wavelet domain inpainting. *IEEE Transactions on Image Processing*, 21(1):106–114, 2011.

[29] X. Zhang and T. F. Chan. Wavelet inpainting by nonlocal total variation. *Inverse Problems and Imaging*, 4(1):191–210, 2010.

[30] S. Zhong. Image coding with optimal reconstruction. In *International Conference on Image Processing*, volume 1, pages 161–164, 1997.