Exact Histogram Specifcation for Digital Images[1] Using a Variational Approach

Mila Nikolova, You-Wei Wen, Raymond Chan

Abstract

We consider the problem of exact histogram specification for digital (quantized) images. The goal is to transform the input digital image into an output (also digital) image that follows a prescribed histogram. Classical histogram modification methods are designed for real-valued images where all pixels have different values, so exact histogram specification is straightforward. Digital images typically have numerous pixels sharing the same value. If one imposes the prescribed histogram to a digital image, usually there are numerous ways of assigning the prescribed values to the quantized values of the image. Therefore, exact histogram specification for digital images is an *ill-posed problem.* So as to satisfy the prescribed histogram, all pixels of the input digital image must be rearranged in a strictly ordered way, while preserving the specific features of the input image. Such a task can be realized if we are able to extract additional *representative* information (called *auxiliary attributes*) from the input digital image. This is a real challenge in exact histogram specification for digital images. We propose a new method that efficiently provides a strict ordering for all pixel values. It is based on a well designed variational approach. Noticing that the input digital image contains quantization noise, we minimize an objective function whose solution is a real-valued image with reduced quantization noise. We show that all the pixels of this real-valued image can be ordered in a strict way with a probability close to one. Then transforming the latter image into another digital image satisfying a specified histogram is an easy task. Numerical results show that our method outperforms by far the preexisting concurrent methods.

Key words: Exact histogram specification, strict-ordering, variational methods, restoration from quantization noise, smooth nonlinear optimization, convex minimization.

I. Introduction

The histogram of an image counts the number of pixels at each intensity value. It is a graph that shows the distribution of the intensity values. Image histogram processing is the act of altering each individual pixel by modifying its dynamic range in order to modify the contrast of the whole image. It is an important image processing task with many real-world applications, such as contrast enhancement [11], [31], [51], segmentation [12], [38], watermarking [15], texture synthesis [46], among many others.

In histogram processing, image intensity level is viewed as a random variable characterized by its probability density function. The histogram of an image shows the empirical distribution of the intensity levels of its pixels. One of the basic histogram processing problem is histogram equalization [25], [44]. It aims to find a

Nikolova is with Centre de Mathématiques et de Leurs Applications, ENS de Cachan, UniverSud, 61 av. du Président Wilson, 94235 Cachan Cedex, France. E-mail: `nikolova@cmla.ens-cachan.fr`.

Wen is with Faculty of Science, Kunming University of Science and Technology, Yunnan, China. The research was supported in part by NFSC Grant. E-mail: `wenyouwei@gmail.com`.

Chan is with Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong. The research was supported in part by HKRGC Grant CUHK400510 and DAG Grant 2060408. E-mail: `rchan@math.cuhk.edu.hk`.

transformation so that the output image has a uniform histogram. In the continuous setting the random variable defined by the cumulative distribution function of the intensity levels is uniformly distributed in $[0, 1]$, and hence such a transformation can always be found. More generally, we may want to yield an output image with pre-specified histogram shapes. This problem is called *histogram specification* or *histogram matching*. The prescribed histogram can be given according to various needs. For example, it can be the histogram of another image, a modified version of the original histogram [45], or a "weighted" histogram of two histograms [17], [18]. In this paper, we do not address how to generate a "good" histogram to improve the input digital image, which is intimately related to perceptual requirements. We restrict our interest to the question: given a histogram, how to specify this histogram exactly.

Numerous methods have been proposed to modify the histogram of an input image. The simplest method is histogram linear stretching [37]. Histogram clipping method [45] limits the maximum number of pixels for each intensity level to a given constant and the clipped pixels are then uniformly distributed among the other intensity levels where the numbers of pixels are less than the clip limit. Several other methods were proposed to preserve the mean brightness of the input image [11], [31], [51]. In [48], Sapiro and Caselles proposed histogram modification via image evolution equations. Arici *et al.* proposed a general framework for histogram modification [1].

The principle behind histogram specification methods is straightforward for real-valued (analog) images: the histogram of the input image and the prescribed histogram should be equalized to uniform distribution first, say by T_i and T_t respectively. Then the output image can be obtained from the composite transformation $T_t^{-1} \circ T_i$. Since the images are real-valued, T_i and T_t are one-to-one functions, and hence $T_t^{-1} \circ T_i$ is well-defined. The principle fails, however, for quantized (digital) images, which is the case of all digital imaging systems. The reason is that for quantized images, the intensity levels of all pixels take a limited number of discrete values. Therefore their cumulative density functions are staircase functions rather than strictly increasing functions like those for the real-valued images.

We will use an example to demonstrate the challenge of the exact histogram specification for discrete image. Consider a 16×16 input image with intensity values living in the set $\{0, 1, 2, 3\}$. The histogram of the input image is shown in Figure 1 (left), the prescribed histogram is shown in Figure 1 (middle). With the classical histogram specification method, the pixels with the same intensity value in the input image should be mapped into the same intensity value. When the classical histogram specification method is applied to the example, the output image is identical to the input one, while the histogram of the specified image is not equal to the prescribed one. From the figure, we observe that the number of pixels with the intensity value "1" in the input image is 96, while the prescribed number in the specified image should be 64. In order to yield an exact histogram specification, the group with intensity value "1" has to be divided into three groups to fill in three bins of the specified histogram: 64 pixels should retain their value, 14 pixels should be mapped to intensity value "0" and 18 pixels to "2". If we do not consider some auxiliary information on pixel values, there exists millions methods to reassign the intensity values.

Methods to obtain strict ordering for a quantized image were proposed to assign the same intensity level to difference intensity levels in [14], [16], [50]. Once all pixels are strictly ordered, the prescribed intensity values are assigned exactly according to the specified histogram. Coltuc *et al*. considered to use the average intensities of neighboring pixels as the auxiliary attribute [16]. Considering two pixels with the same intensity value, the mean values over the neighborhoods centered on each pixel are compared to order these two pixels. If the mean values are still the same, then they choose larger neighborhoods and continue in the same way until all pixels are ordered. Wan and Shi argued that the local mean approach fails to sharpen the edges of the output image [50]. They proposed to order the pixels according to the absolute values of its wavelet

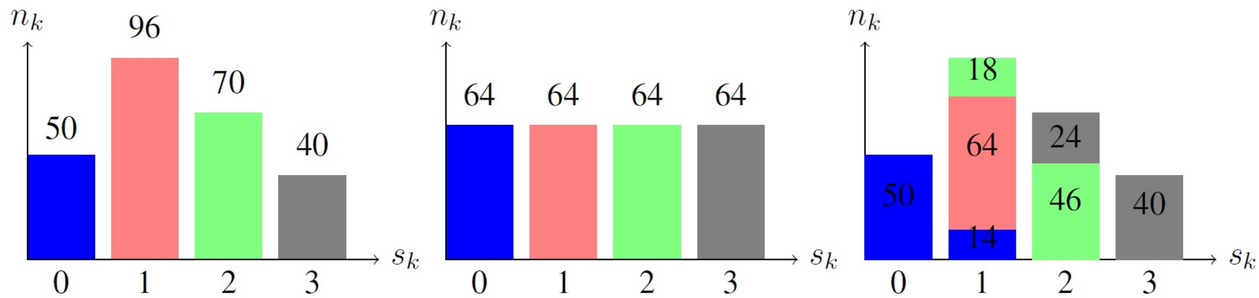


Fig. 1: Left: histogram of the input image; middle: prescribed histogram; right: remapping the groups of pixels with the same intensity value to pixels with different intensity values.

coefficients. The wavelet-based approach tends to amplify the noise since a noise in a smooth region may be mistaken as an edge and hence is sharpened. Post-processing approach or iterative methods can be applied to suppress the amplified noises [3]. We emphasize that both the local mean approach and the wavelet-based approach cannot realize strict ordering without degrading the input quantized image. This is a major drawback.

In this paper, we present a variational method that enables us to strictly order the pixel values of a quantified image by restoring it from the quantization noise. We give a theoretical analysis of the method and prove that the pixels of the restored image can be ordered in a strict way with a probability close to one. A sketch of some of the ideas of our approach was given in a conference paper [10]. Here we present a lot of experimental results showing that the proposed method is very efficient and produces images of better quality than both the local mean method [16] and the wavelet-based method [50].

The outline of the paper is as follows. In Section II, we give sorting algorithm for exact histogram specification. In Section III, we present the proposed variational method. In Section IV, we summarize the algorithm for exact histogram specification. In Section V, numerical examples are given to demonstrate the effectiveness of the proposed model. Concluding remarks are given in Section VI.

II. Sorting Algorithms

Let $\mathbf{u}$ be an M-by-N image obtained by digitizing an analog image $\mathbf{u}_o$ (on a discrete grid) with range in some interval $[0, a)$. We assume the possible quantized values that $\mathbf{u}$ can take are from

$$\mathcal{P} \stackrel{\text{def}}{=} \{p_1, \cdots, p_L\} \tag{1}$$

and that p_k are in increasing order. For 8-bit images, $\mathcal{P} = \{0, \cdots, 255\}$. In the following we will express $\mathbf{u}$ as an n-vector by concatenating the columns in $\mathbf{u}$. Here $n = MN$. Denote $\mathbb{I}_q \stackrel{\text{def}}{=} \{1, \ldots, q\}$ for any positive integer q and $\Omega_k \stackrel{\text{def}}{=} \{i \in \mathbb{I}_n \ \mid \ \mathbf{u}[i] = p_k\}$, $k = 1, 2, \cdots, L$. The associated histogram of $\mathbf{u}$ is the L-tuple $\mathbf{h}_\mathbf{u} = (|\Omega_1|, |\Omega_2|, \ldots, |\Omega_L|)$, where $|\cdot|$ stands for cardinality. The image $\mathbf{u}$ with histogram $\mathbf{h}_\mathbf{u}$ is a result of quantization of the original real-valued image $\mathbf{u}_o$. This amounts to set to the same intensity level p_k all the values of $\mathbf{u}_o$ on the interval $[t_{k-1}, t_k)$, $k = 1, \ldots, L$. Then $|\Omega_k| = n \int_{t_{k-1}}^{t_k} p(\mathbf{u}_o) dt$, where $p(\mathbf{u}_o)$ is the empirical probability density function of $\mathbf{u}_o$, and $t_0 = 0$ and $t_L = a$.

Let the pre-specified histogram read $\mathbf{h} = (h_1, h_2, \ldots, h_L)$. The classical way of defining exact histogram specification is designed for real-value images like $\mathbf{u}_o$. Such images are nowhere constant, see [26]. In such a case, exact histogram specification can be summarized as follows: a sequence $\bar{t}_1, \bar{t}_2, \ldots, \bar{t}_{L-1}$ is computed such that $h_k = n \int_{\bar{t}_{k-1}}^{\bar{t}_k} p(\mathbf{u}_o) dt$ for $k = 1, 2, \ldots, L-1$ with $\bar{t}_0 = 0$ and $\bar{t}_L = a$. Once such sequence $\{\bar{t}_k\}$ is obtained, the intensity values in the interval $[\bar{t}_{k-1}, \bar{t}_k)$ are assigned the intensity of p_k, and we are done.

However, in practice, we only have the digitized image $\mathbf{u}$, obtained from an original real-valued (analog) image $\mathbf{u}_o$. The digital $\mathbf{u}$ can seldom be totally-ordered—its pixels have only L possible magnitudes. Thus, the number of pixels in $\mathbf{u}$ is generally much larger than the number of intensity levels, i.e., $n \gg L$. It is therefore very likely that some Ω_k, $k = 1, \dots, L$, will have $|\Omega_k| > h_k$. In order to satisfy the prescribed histogram, some pixels in Ω_k will have values mapped to other intensity levels. The number of ways of selecting the pixels to assign to other intensity levels is very large. Consequently, this problem is ill-posed.

The key to a stable solution is to create a total ordering of the pixels in the same Ω_k by learning some auxiliary information from the digital image $\mathbf{h}$ itself. Suppose that for any pixel $i \in \mathbb{I}_n$, we can create $K-1$ auxiliary information $\kappa_1[i], \dots, \kappa_{K-1}[i]$. Then we can define an ascending ordering "'$\prec$" for all pixels in $\mathbb{I}_n$ based on the K-tuples $(\mathbf{u}[i], \kappa_1[i], \dots, \kappa_{K-1}[i])$. To facilitate the discussions, let $\kappa_0[i] \stackrel{\text{def}}{=} \mathbf{u}[i]$. For any two pixels i and j in $\mathbb{I}_n$, we say that $i \prec j$ if for some $0 \le \ell \le K-1$

$$\kappa_s[i] = \kappa_s[j] \text{ for all } \; 0 \le s \le \ell - 1 \text{ and } \kappa_\ell[i] < \kappa_\ell[j].$$

For good choices of auxiliary information and K sufficiently large, one can in principle sort all pixels i in $\mathbb{I}_n$ according to the ordering $\prec$. That is, we can order the pixels i in $\mathbb{I}_n$ in such a way that $i_1 \prec i_2 \prec \ldots \prec i_n$. Once such a strict-ordering is obtained, matching the input histogram to the prescribed one is straightforward. This can be done by dividing the ordered list $\{i_\ell\}_{\ell=1}^n$ from left to right into L groups. Thus the first h_1 pixels $i_1, i_2, \dots, i_{h_1}$ belong to the first group, and are assigned the intensity of p_1. The next h_2 pixels $i_{h_1+1}, \dots, i_{h_1+h_2}$ belong to the second group and are assigned the intensity of p_2, and so on until all pixels are assigned to their new intensities.

Several ideas have been proposed for the auxiliary information. Coltuc *et al.* proposed to use the local average intensities of a pixel's neighborhood as auxiliary information [16]. For pixels having the same intensity, if the average intensities of their neighborhoods are the same, then a larger neighborhood will be chosen to compute the average intensity. This procedure is repeated until all pixels are ordered. The author claimed that $K = 6$ is appropriate for any application. Wan and Shi proposed to order the pixels according to the absolute values of the wavelet coefficients of the whole image [50]. In next section, we present a new approach to obtain pertinent auxiliary information in $\mathbf{u}$. Using a variational approach, we create an intermediate image $\widehat{\mathbf{f}}$ whose pixels can be totally ordered with probability one.

III. A Variational Approach

Our approach to obtain auxiliary information is based on a different paradigm. The available image $\mathbf{u}$ contains quantization noise. Our strategy is to built a real-valued image which removes as much as possible this quantization noise. Even though the original real $\mathbf{u}_o$ is unavailable, some general priors such as the presence of edges and fine structures in $\mathbf{u}_o$ can be employed to produce such a restored version of $\mathbf{u}$. A subtle task like this can be handled by the means of a well conceived variational method.

Given $\mathbf{u}$, its real-valued restoration, denoted by $\widehat{\mathbf{f}}$, is defined by

$$\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u}) = \min_{\mathbf{f} \in \mathbb{R}^n} \mathcal{J}(\mathbf{f}, \mathbf{u}),$$

where $\mathcal{J}(\cdot, \mathbf{u}) : \mathbb{R}^n \to \mathbb{R}$ is a regularized convex cost function of the form

$$\mathcal{J}(\mathbf{f}, \mathbf{u}) = \Psi(\mathbf{f}, \mathbf{u}) + \beta \Phi(\mathbf{f}), \tag{2}$$

combining a data fitting term Ψ and a regularization term Φ, weighted by a parameter $\beta > 0$. These terms are given by

$$\begin{aligned} \Psi(\mathbf{f}, \mathbf{u}) &= \sum_{i \in \mathbb{I}_n} \psi(\mathbf{f}[i] - \mathbf{u}[i]), && (3) \\ \Phi(\mathbf{f}) &= \sum_{i \in \mathbb{I}_n} \sum_{j \in \mathcal{N}_i} \phi(\mathbf{f}[i] - \mathbf{f}[j]). && (4) \end{aligned}$$

In (4), $\mathcal{N}_i$ is the set of the four or the eight adjacent neighbors of pixel i in the image, for every $i \in \mathbb{I}_n$. We consider either Neumann or mirror boundary conditions on the set $\{\mathcal{N}_i \mid i \in \mathbb{I}_n\}$. Pivotal condition on the real scalar functions ψ and ϕ in (3)–(4) are described in H1 and in Q1 below.

H1. *The functions $\phi : \mathbb{R} \to \mathbb{R}$ and $\psi : \mathbb{R} \to \mathbb{R}$ are $\mathcal{C}^s$ for $s \geq 2$, symmetric and satisfy*

$$t \in \mathbb{R} \quad \Rightarrow \quad \phi''(t) > 0 \ \ \textit{and} \ \ \psi''(t) > 0 \ .$$

In order to provide some quantization noise removal, it is required that $\mathcal{J}(\cdot, \mathbf{u})$ is *detail-preserving*. To this end, both functions ψ and ϕ should satisfy a qualitative requirement:

Q1. *Functions ψ and ϕ are nearly affine beyond a small neighborhood of the origin.*

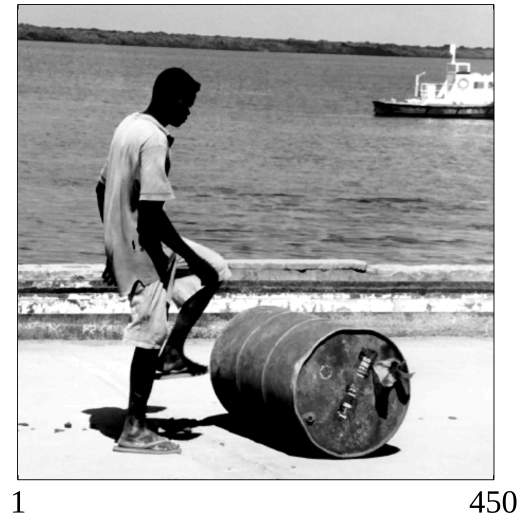
Good choices for ψ and ϕ, meeting H1 and Q1, are given in Table I. Customarily, such functions are involved only in the regularization term, see e.g. [7], [2]. So as to achieve our objectives, they are pertinent to define the data fitting term as well.

	ψ, ϕ	ψ'', ϕ''
f1	$\left(t^2 + \alpha\right)^{1/2}$	$\left(t^2 + \alpha\right)^{-3/2}$
f2	$\log(\cosh(\alpha t))$	$\alpha^2(\cosh(\alpha t))^{-2}$
f3	$\lvert t\rvert/\alpha - \log\left(1 + \lvert t\rvert/\alpha\right)$	$(\alpha + \lvert t\rvert)^{-2}$

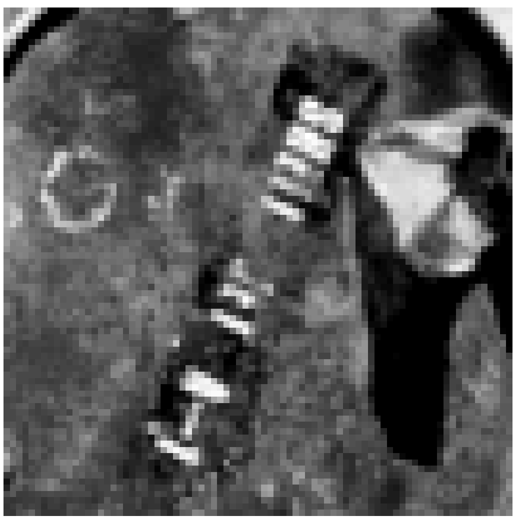
TABLE I: Relevant choices for ψ and ϕ obeying H1 and Q1. The size of the neighborhood of zero where these functions are not "nearly affine" is controlled by the parameter $\alpha > 0$.

a) Motivation to choose a $\mathcal{J}$ satisfying H1 and Q1.: Before to go into the details, we explain the intuition behind the demands H1 and Q1 addressed to $\mathcal{J}$ in (2). Our main concern is to obtain a restoration $\widehat{\mathbf{f}}$ of $\mathbf{u}$ whose pixels are all different from each other while being close to $\mathbf{u}$ but "better" than $\mathbf{u}$. Since ϕ satisfies H1, we will get minimizers $\widehat{\mathbf{f}}$ that are generically nowhere nonconstant [41]. Indeed, natural images have been shown to be almost nowhere constant [26]. Rather, they present large variations, edges and fine structures. The trend of the large variations is available in the input image $\mathbf{u}$. Requirement Q1 on ϕ enables the recovery of edges and details and in this way some removal of the quantization noise. For instance, $\alpha > 0$ in f1, Table I, should be just small enough. Selecting a small $\beta > 0$ (compared to the range of $\mathbf{u}$) enhances fitting to data $\mathbf{u}$. If the neighborhood near the origin where ψ is not quasi-affine goes to zero (e.g. α is almost zero in f1, Table I), ψ tends to the absolute value function. The latter is known to generate minimizers $\widehat{\mathbf{f}}$ containing a certain number of entries equal to the relevant entries of $\mathbf{u}$ [40]. Such minimizers may still contain numerous equally valued pixels, hence such a scenario must be avoided. If ψ obeys H1 and Q1 (e.g., for f1, Table I, we should take $\alpha > 0$ very small), the components of $\widehat{\mathbf{f}}$ will certainly be close but different from the relevant entries of $\mathbf{u}$. So the level sets of the digital image $\mathbf{u}$ are not destroyed, but only refined. This is the reason why we can say that details are preserved in $\widehat{\mathbf{f}}$. Undoubtedly, pixels in $\widehat{\mathbf{f}}$ must change from those in $\mathbf{u}$ no more than a given value (for example $|0.5|$). The soundness of these arguments is illustrated in Fig. 2.

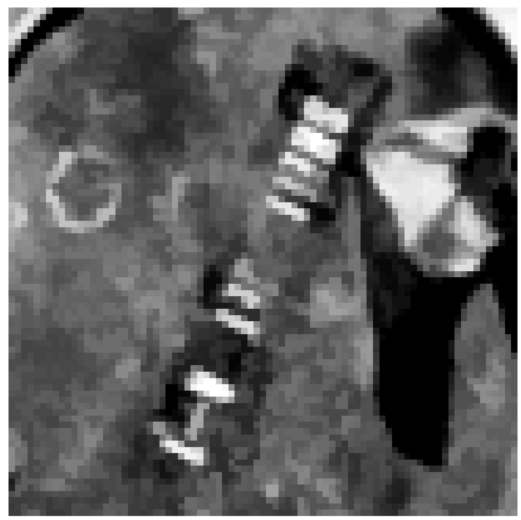
(a) Original digital image: the pixels with value 249 are marked with red dots



(b) Restored image $\beta = 0.2$, $\psi = \phi =$ f1 (Table I), $\alpha = 0.05$



(c) 80×80 zoom of the original image



(d) 80×80 zoom of the restored image

Fig. 2: The original digital image is of size 450×450 and the values of its pixels belong to the set $\{0, \cdots, 255\}$. The restored image $\widehat{\mathbf{f}}$ is obtained by minimizing $\mathcal{J}$ where $\psi(t) = \sqrt{\alpha_1 + t^2}$ and $\phi(t) = \sqrt{\alpha_2 + t^2}$. For $\alpha_1 = \alpha_2 = 0.05$ and $\beta = 0.1$, all pixels of $\widehat{\mathbf{f}}$ have different values, so they can be sorted in a strict way. The restored image plotted here corresponds to $\alpha_1 = \alpha_2 = 0.05$ and $\beta = 0.2$ in order to enhance the restoration effect. Its pixels can be sorted in a strict way as well. The zooms on the second row show that the restored image has a much more regular appearance than the digital one, so quantization noise was reduced.

Below we exhibit the salient properties of the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ under H1. It worths noticing that they are irrelevant to the qualitative requirement Q1, as well as to the particular shapes of ψ and ϕ and to the choice of $\beta > 0$. The latter remark presents a challenging topic for future research.

A. Preliminary results

The differences $\{\mathbf{f}[i]-\mathbf{f}[j] \mid j \in \mathcal{N}_i\}$ for all $i \in \mathbb{I}_n$ in (4) can be rewritten using finite difference operators $\mathbf{g}_i \in \mathbb{R}^n$, $1 \le i \le r$, where r is the total number of these operators. Then Φ reads

$$\Phi(\mathbf{f}) = \sum_{i\in\mathbb{I}_r} \phi(\mathbf{g}_i^T \mathbf{f}) \ . \tag{5}$$

Let us denote

$$G = \begin{bmatrix} \mathbf{g}_1^T \\ \vdots \\ \mathbf{g}_r^T \end{bmatrix} \in \mathbb{R}^{r\times n}.$$

According to (4) and the adopted boundary conditions (Neumann or mirror), we have $\ker G = \{c\mathbb{1} \mid c \in \mathbb{R}\}$, where $\mathbb{1}$ is a vector composed of ones.

We shall study how the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ behaves as a function of data $\mathbf{u}$. This goal motivates the definition below which was originally introduced in [39]. For clarity, it is restated in a way adapted to this work.

Definition 1. *A function $\mathcal{F} : \mathcal{O} \to \mathbb{R}^n$, where $\mathcal{O}$ is an open domain in $\mathbb{R}^n$, is said to be a* minimizer function *relevant to the family of functions $\mathcal{J}(\cdot, \mathcal{O})$ if for every $\mathbf{u} \in \mathcal{O}$, the point $\widehat{\mathbf{f}} = \mathcal{F}(\mathbf{u})$ is a strict local minimizer of $\mathcal{J}(\cdot, \mathbf{u})$.*

The next lemma is a straightforward extension of the Implicit Functions Theorem [4]. Its proof can be found e.g. in [21, Theorem 6, p. 34] or in [30, Lemma 6.1.1, p. 268]. In what follows, D_i^j stands for the jth order differential of a function with respect to the ith variable[1].

Lamma 1. *Suppose that $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is any function which is $\mathcal{C}^s$, with $s \ge 2$. Fix $\mathbf{u} \in \mathbb{R}^n$. Let $\widehat{\mathbf{f}} \in \mathbb{R}^n$ be such that $D_1\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u}) = 0$ and $D_1^2\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u})$ is positive definite. Then there exists an open neighborhood of $\mathbf{u}$, say $\mathcal{O}$, and a unique $\mathcal{C}^{s-1}$-function $\mathcal{F} : \mathcal{O} \to \mathbb{R}^n$ such that $\mathcal{F}(\mathbf{u}) = \widehat{\mathbf{f}}$.*

We will see that in our case, the minimizer function is uniquely defined on $\mathbb{R}^n$.

Proposition 1. *Let $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ in (2) satisfy H1. Then for any $\beta > 0$, $\mathcal{J}$ has a unique minimizer function $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ which is $\mathcal{C}^{s-1}$ continuous.*

The proof of the proposition is given in Appendix VII-A. The components of the minimizer function $\mathcal{F}$ read $\mathcal{F}_i$, $i \in \mathbb{I}_n$. A diagonal matrix A with diagonal entries $a[i]$, $i \in \mathbb{I}_n$, is denoted by $A = \mathrm{diag}\big(\{a[i]\}_{i=1}^n\big)$.

Lamma 2. *Let $\beta > 0$ be arbitrary and $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ in (2) satisfy H1. Then its Hessian matrix $H(\mathbf{u}) \overset{\text{def}}{=} D_1^2\mathcal{J}(\mathcal{F}(\mathbf{u}), \mathbf{u})$, given by*

$$H(\mathbf{u}) = \mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u}) - \mathbf{u}[i])\}_{i=1}^n\big) + \beta G^T \mathrm{diag}\big(\{\phi''(\mathbf{g}_i^T \mathcal{F}(\mathbf{u}))\}_{i=1}^r\big) G \tag{6}$$

is invertible. Consequently, the differential $D\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ of the minimizer function $\mathcal{F}$ of $\mathcal{J}$ reads

$$D\mathcal{F}(\mathbf{u}) = \begin{bmatrix} D\mathcal{F}_1(\mathbf{u}) \\ \cdots \\ D\mathcal{F}_p(\mathbf{u}) \end{bmatrix} = \big(H(\mathbf{u})\big)^{-1} \mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u}) - \mathbf{u}[i])\}_{i=1}^n\big)$$

and satisfies

$$\mathrm{rank}\,(D\mathcal{F}(\mathbf{u})) = n, \quad \forall \mathbf{u} \in \mathbb{R}^n. \tag{7}$$

[1]E.g., $D_1^j\mathcal{J}$ is the jth differential of $\mathcal{J}$ in (2) with respect $\mathbf{f}$ and $D_2^j\mathcal{J}$—with respect to $\mathbf{u}$.

This easy lemma, outlined in Appendix VII-B, underlies the main theoretical results established in this work.

B. The crucial feature of the minimizers of $\mathcal{J}(\cdot,\mathbf{u})$ and discussion

The set $\mathcal{G}$ given next is composed of all operators that yield the difference between any two pixels in an image:

$$\mathcal{G} \stackrel{\text{def}}{=} \bigcup_{(i,j)\in\mathbb{I}_n\times\mathbb{I}_n} \left\{ g\in\mathbb{R}^n \ \mid \ g[i]=-g[j]=1,\ i\neq j,\ (i,j)\in\mathbb{I}_n,\ g[k]=0,\ k\in\mathbb{I}_n\setminus(i\cup j)\right\}\ . \tag{8}$$

Hence all difference operators in (5) satisfy $\mathbf{g}_i\in\mathcal{G},\forall i\in\mathbb{I}_r$. As usual, the Lebesgue measure in $\mathbb{R}^n$ is denoted by $\mathbb{L}^n(\cdot)$. Our main result is stated below. Its proof is presented in Appendix VII-C.

Theorem 1. *Let $\mathcal{J}:\mathbb{R}^n\times\mathbb{R}^n\to\mathbb{R}$ in* (2) *satisfy H1. For its minimizer function $\mathcal{F}:\mathbb{R}^n\to\mathbb{R}^n$, define the set $K_{\mathcal{G}}$ as it follows:*

$$K_{\mathcal{G}}=\bigcup_{g\in\mathcal{G}}\left\{\mathbf{u}\in\mathbb{R}^n \ \mid \ g^T\mathcal{F}(\mathbf{u})=0\right\}. \tag{9}$$

Then $K_{\mathcal{G}}$ is closed in $\mathbb{R}^n$ and obeys

$$\mathbb{L}^n(K_{\mathcal{G}})=0\ .$$

The result holds true for any $\beta>0$ in $\mathcal{J}$.

The set $K_{\mathcal{G}}$ in (9) contains all possible $\mathbf{u}\in\mathbb{R}^n$ such that a minimizer $\widehat{\mathbf{f}}=\mathcal{F}(\mathbf{u})$ of $\mathcal{J}(\cdot,\mathbf{u})$ might have two equal entries, $\widehat{\mathbf{f}}[i]=\widehat{\mathbf{f}}[j]$ for some $i\neq j$ where $(i,j)\in\mathbb{I}_n\times\mathbb{I}_n$.

The elements of $K_{\mathcal{G}}$ are highly exceptional in $\mathbb{R}^n$. Such data cannot hence give rise to a minimizer having two equal entries. Indeed, the subset $\mathbb{R}^n\setminus K_{\mathcal{G}}$ contains an *open and dense* subset of $\mathbb{R}^n$. The chance that a truly random $\mathbf{u}\in\mathbb{R}^n$—i.e. a $\mathbf{u}$ following a non-singular probability distribution on $\mathbb{R}^n$—comes across such an $K_{\mathcal{G}}$ can be ignored in practice. Conversely, $\mathcal{F}_i(\mathbf{u})\neq\mathcal{F}_j(\mathbf{u})$, for $i\neq j$, is a *generic* property of the minimizers $\mathcal{F}$ of $\mathcal{J}:\mathbb{R}^n\times\mathbb{R}^n$, as given in (2) and satisfying H1 (using the terminology in [19]).

b) Discussion: Let us denote by $\mathcal{S}^n_{\mathcal{P}}$ the set of all $M\times N$ images whose $n=MN$ pixels values belong to the quantization set $\mathcal{P}$ in (1):

$$\mathcal{S}^n_{\mathcal{P}}=\left\{u\in\mathbb{R}^n \ \mid \ u[i]\in\mathcal{P},\ \ \forall i\in\mathbb{I}_n\right\}\ . \tag{10}$$

Reminding that $|\mathcal{P}|=L$, the cardinality of $\mathcal{S}^n_{\mathcal{P}}$ is $|\mathcal{S}^n_{\mathcal{P}}|=L^n$. Even though $|\mathcal{S}^n_{\mathcal{P}}|$ is a *huge* number[2], the set $\mathcal{S}^n_{\mathcal{P}}$ is finite, hence

$$\mathbb{L}^n(\mathcal{S}^n_{\mathcal{P}})=0\ .$$

Hence the hot question: *what can we say about the possible intersection between $K_{\mathcal{G}}$ and $\mathcal{S}^n_{\mathcal{P}}$?*

As in practice, assume that $\mathcal{P}$ is composed of L integers. Let $N_{\mathcal{P}}$ be the set of the functions that map $\mathcal{P}^n$ onto $\mathcal{P}$. We would not like that the minimizer function $\mathcal{F}$ has some components belonging to $N_{\mathcal{P}}$. Number theory gives limited answers to the question of the kind of functions being able to come across $N_{\mathcal{P}}$. For $n=1$, every function applied to an integer u and yielding an integer $\hat{u}$ is of the form

$$f(u)=\sum_{i\in\mathbb{I}_k} b_i \begin{pmatrix} u\\ i\end{pmatrix} \quad \text{where} \quad \begin{pmatrix} u\\ i\end{pmatrix}=\frac{u(u-1)\cdots(u-i+1)}{i\,!} \quad \text{and} \quad u\leq i-1,$$

[2]For 512×512, 8-bits images, this value is 255^{512^2}—the amount of all 512×512, 8-bits pictures that people can ever take with their digital cameras.

where *all b_i are integers*. The question was initially posed in [43]. This result along with some refinements can be found in [8]. However, $f(u) \in N_{\mathcal{P}}$ requires also that $f(u) \in \mathcal{P}$, which *drastically limits*[3] the functions of this form that fall into $N_{\mathcal{P}}$. In the case of several variables, some polynomial functions may belong to $N_{\mathcal{P}}$, see [9]. More generally, Diophantine equations [27] can also be cast in the class of polynomial functions. Under the (severe) restriction $f(\mathbf{u}) \in \mathcal{P}$, some of them also live in $N_{\mathcal{P}}$. To the best of our knowledge, no other families of functions were exhibited to be able to cross the subset $N_{\mathcal{P}}$.

However, given the expression for $D\mathcal{F}$ in Lemma 2, it is not difficult to see that no component $\mathcal{F}_i$ of our minimizer function $\mathcal{F}$ can have a polynomial expression.

Let $\mathbf{u} = c\mathbb{1}$ for some $c \in \mathcal{P}$. Then for any $\beta > 0$, the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}(\cdot, \mathbf{u})$ reads $\widehat{\mathbf{f}} = \mathbf{u}$. Indeed, since $c\mathbb{1} \in \ker G$, we have $\Psi(\mathbf{u}, \mathbf{u}) = 0$ and $\Phi(\mathbf{u}) = \beta r \phi(0)$, so $\mathcal{J}$ reaches its lower bound for $\widehat{\mathbf{f}} = \mathbf{u}$. Hence all constant digital images meet

$$\{c\mathbb{1} \ \mid \ c \in \mathcal{P}\} \in K_{\mathcal{G}} \cap \mathcal{S}^n_{\mathcal{P}} \ .$$

Consequently,

$$K_{\mathcal{G}} \cap \mathcal{S}^n_{\mathcal{P}} \neq \varnothing \ .$$

One can ask what histogram modification would be needed for a constant image. However, there may be other simple images that belong to $K_{\mathcal{G}} \cap \mathcal{S}^n_{\mathcal{P}}$. We can reasonably conjecture the following:

- $K_{\mathcal{G}} \cap \mathcal{S}^n_{\mathcal{P}}$ is *essentially* composed of simple (synthetic, in practice) images. For most of them, if some histogram modification was needed, it should be defined in a proper way (see Remark 2 concerning the synthetic image in Fig. 3, p. 11).
- The ratio
$$\frac{\left|\left(K_{\mathcal{G}} \cap \mathcal{S}^n_{\mathcal{P}}\right)\right|}{|\mathcal{S}^n_{\mathcal{P}}|}$$
should be a number close to zero.

Being impossible to prove this conjecture, we give additional theoretical defence to support our approach, as well as tests showing its numerical evidence in section V.

C. More arguments in favor of our approach

We wish to know if some entries of a minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}(\cdot, \mathbf{u})$ can take a value equal to some components of the data image $\mathbf{u}$ or belonging to the quantization set $\mathcal{P}$.

Proposition 2. *Let $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ in* (2) *satisfy H1. Then for its minimizer function $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$*

(i) *the set $K_{\mathcal{I}}$ given below*
$$K_{\mathcal{I}} = \bigcup_{i \in \mathbb{I}_n} \bigcup_{i \in \mathbb{I}_n} \left\{\mathbf{u} \in \mathbb{R}^n \ \mid \ \mathcal{F}_i(\mathbf{u}) = \mathbf{u}[j]\right\} \ , \tag{11}$$
is closed in $\mathbb{R}^n$ and obeys $\mathbb{L}^n(K_{\mathcal{I}}) = 0$;

(ii) *the set $K_{\mathcal{P}} = \bigcup_{p \in \mathcal{P}} \bigcup_{i \in \mathbb{I}_n} \left\{\mathbf{u} \in \mathbb{R}^n \ \mid \ \mathcal{F}_i(\mathbf{u}) = p\right\}$, where $\mathcal{P}$ is the quantization set introduced in* (1)*, is closed in $\mathbb{R}^n$ and meets $\mathbb{L}^n(K_{\mathcal{P}}) = 0$.*

These statements are valid for any $\beta > 0$ involved in $\mathcal{J}$.

[3] These polynomial functions yield also arbitrarily large values that exceed the bounded set $\mathcal{P}$.

The proof of these statements can be found in Appendix VII-D.

The set $K_{\mathcal{I}}$ in (11) contains all possible $\mathbf{u} \in \mathbb{R}^n$ such that the minimizer $\widehat{\mathbf{f}} = \mathcal{F}(\mathbf{u})$ might contain some entries equal to data entries, $\mathcal{F}_i(\mathbf{u}) = \mathbf{u}[j]$ for some $(i, j) \in \mathbb{I}_n \times \mathbb{I}_n$. In particular, the event $\mathcal{F}_i(\mathbf{u}) = \mathbf{u}[i]$ is highly exceptional, as anticipated in the paragraph Motivation next to H1. According to Proposition 2 (i), the minimizer functions $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ relevant to $\mathcal{J}$ in (2), *generically* satisfy $\mathcal{F}_i(u) \neq \mathbf{u}[j]$, for all $(i, j) \in \mathbb{I}_n \times \mathbb{I}_n$. This constitutes a general result holding for any real data $\mathbf{u} \in \mathbb{R}^n$.

Statement (ii) is more specialized: in a generic sense, no entry of a minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}(\cdot, \mathbf{u})$ can take a quantized value belonging to $\mathcal{P}$. Yet again, the pleading exposed in paragraph *Discussion* following Theorem 1 remains actual.

Proposition 3. *Let for an arbitrary $\beta > 0$, the cost function $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ in* (2) *satisfy H1. Then its minimizer function $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ is nonexpansive:*

$$(\mathbf{u}, \boldsymbol{\zeta}) \in \mathbb{R}^n \times \mathbb{R}^n \quad \Rightarrow \quad \|\mathcal{F}(\mathbf{u} + \boldsymbol{\zeta}) - \mathcal{F}(\mathbf{u})\|_2 \leq \|\boldsymbol{\zeta}\|_2 \ . \tag{12}$$

The proof, given in Appendix VII-F, uses a lemma which is stated and proven in Appendix VII-E.

The fact that the minimizer function $\mathcal{F}$ is non-expansive supports our conjecture as well: typically, the values of the input digital image $\mathbf{u}$ are slightly reduced at the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$.

After all, we cannot *guarantee* that for *absolutely any digital image* $\mathbf{u} \in \mathcal{P}^n$, the entries of the minimizer $\widehat{\mathbf{f}} = \mathcal{F}(\mathbf{u})$ can be ordered in a strictly increasing way.

What to do if ever we have found equally valued pixels in $\widehat{\mathbf{f}}$? The theorem enounced below can help.

Theorem 2. *Let $\mathcal{J} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ in* (2) *satisfy H1. Then its minimizer function $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ is differentiable with respect to β since*

$$\frac{d\mathcal{F}(\mathbf{u})}{d\beta} = -\big(H(\mathbf{u})\big)^{-1} D\Phi(\mathcal{F}(\mathbf{u})) \ ,$$

where $H(\mathbf{u})$ is the Hessian matrix given in (6). *More precisely, using the original expression for Φ in* (4), *each component k of the n-length vector $\frac{d\mathcal{F}(\mathbf{u})}{d\beta}$ reads*

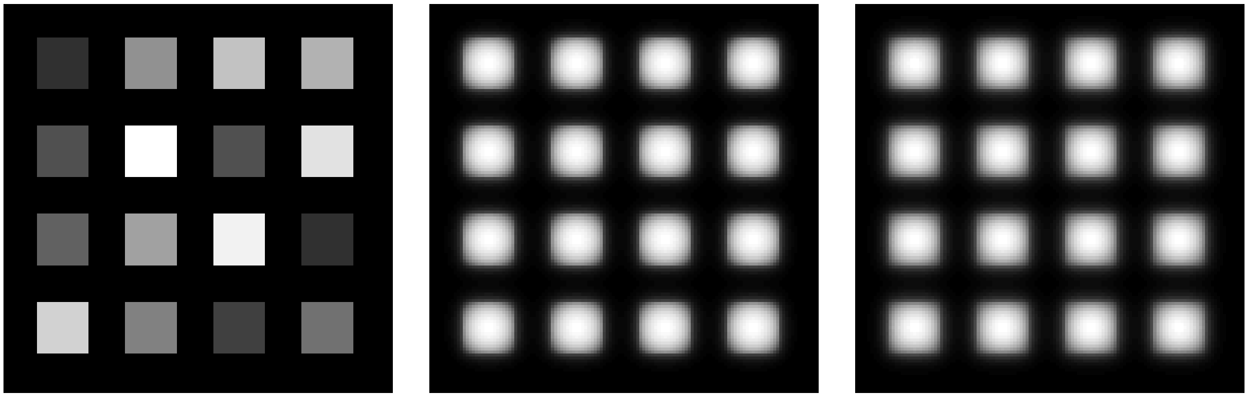
$$\frac{d\mathcal{F}_k(\mathbf{u})}{d\beta} = -2 \sum_{i \in \mathbb{I}_n} \big(H(\mathbf{u})\big)^{-1}[k, i] \sum_{j \in \mathcal{N}_i} \phi'(\mathcal{F}_i(\mathbf{u}) - \mathcal{F}_j(\mathbf{u})) \quad \forall k \in \mathbb{I}_n \ , \tag{13}$$

where $\big(H(\mathbf{u})\big)^{-1}[k, i]$ are the coefficients of the matrix $\big(H(\mathbf{u})\big)^{-1}$.

This statement, demonstrated in Appendix VII-G, shows that pixel values change continuously with $\beta > 0$. Let us mention that one can also prove under reasonable conditions that $\big(H(\mathbf{u})\big)^{-1}[k, i] \geq 0$, $\forall (k, i) \in \mathbb{I}_n \times \mathbb{I}_n$. For a given β, if ever we had some equally valued pixels in $\widehat{\mathbf{f}}$, this means that the data fidelity term is still too dominant. In such a case, *we should just slightly increase* β and compute the relevant minimizer. According to Theorem 2, the new minimizer will change a bit in decreasing the importance of the data term. An illustration on a synthetic digital image is provided in Fig. 3.

The last expression (13) suggests that a larger neighborhood (the eight rather than the four adjacent pixels) might produce a more significant change for a smaller change of β. This is corroborated by Fig. 3, see also the explanation given in caption. However, the computational effort to get $D\mathcal{J}$ is nearly twice more important that for 4 neighbors. As far as Theorem 2 tells us that in both cases, if for some β we had a minimizer $\widehat{\mathbf{f}}$ containing some pixels with equal values, an increase of β will produce a minimizer whose pixels can be sorted in a strict way. Therefore, it appears more reasonable to *use in practice only the four nearest neighbors* in the expression for Φ in (4).

Remark 1. *Even though the set $K_{\mathcal{G}}$ in Theorem 1 and the sets $K_{\mathcal{I}}$ and $K_{\mathcal{P}}$ in Proposition 2 are closed and of null Lebesgue measure in $\mathbb{R}^n$ for any $\beta > 0$, changing β modifies their content, hence their intersection with the set $S^n_{\mathcal{P}}$ in* (10) *of all digital images with n pixels and values in $\mathcal{P}$ changes as well.*



(a) Synthetic digital image $\mathbf{u}$

(b) The minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ for $\beta = 2$ and $|\mathcal{N}_i| = 8$: **all pixels are different**

(c) The minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ for $\beta = 5$ and $|\mathcal{N}_i| = 4$: **all pixels are different**

Fig. 3: The original image (128×128) in (a) has a null background. All squares are constant and their values belong to $\{48, 64, 80, 96, 128, 144, 160, 176, 192, 208, 224, 240, 256\}$. We consider $\mathcal{J}$ for $\psi(t) = \sqrt{\alpha_1 + t^2}$ and $\phi(t) = \sqrt{\alpha_2 + t^2}$ where $\alpha_1 = \alpha_2 = 0.1$. For $\beta = 2$, if Φ is defined using the four adjacent neighbors (i.e $|\mathcal{N}_i| = 4$), the minimizer $\widehat{\mathbf{f}}$ involves 12.2% pairs of equal pixels whereas if $|\mathcal{N}_i| = 8$, all pixels of $\widehat{\mathbf{f}}$ are different from each other—see (b). For $\beta = 5$, in both cases, $|\mathcal{N}_i| = 4$ and $|\mathcal{N}_i| = 8$—see (c), we obtain a minimizer $\widehat{\mathbf{f}}$ whose pixels can be sorted strictly.

Remark 2. *For the original image in Fig. 3, a reasonable histogram modification would be to change the values of the squares. For this purpose, sorting the pixels in a strict way would clearly be a bad approach. In the same way, if a real-world digital image contains quite an extended constant region, a reasonable histogram modification should keep it constant.*

IV. Numerical scheme

A. Some practical limitations

In spite of the theory presented in section III, it can occur that for a given β, the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ contains some equally valued pixels. Several important reasons are mentioned next.

- The real numbers that a computer treats are in fact just a large but finite set of numbers. E.g. Matlab cannot distinguish numbers smaller than 2.2×10^{-16}.
- The theory supposes that we deal with *exact* minimizers of $\mathcal{J}$. However, in practice we cannot get such minimizers. It worths emphasizing that a $\mathcal{J}$ satisfying H1 and Q1 contains large nearly flat regions, so its minimization is not easy. The algorithm being initialized with the digital input image $\mathbf{u}$, an *inexact* minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ might contain pixels with equal values. Indeed, we have observed that in such cases, increasing the precision of the minimization method usually enables to sort in a strict way all pixels of $\widehat{\mathbf{f}}$.

- If the digital image contains large constant regions (see Remark 2) or if it involves several equal patterns having the same background (this typically can arise for images having a low compression rate), it is clear that for a small β the resultant $\widehat{\mathbf{f}}$ will have pixels sharing the same value.
- Last, we remind that the set $K_{\mathcal{G}} \cup \mathcal{S}_{\mathcal{P}}^n$ is not empty (see also Remark 1).

All these practical reasons show that in general we need a tool to improve the sorting of the pixels of $\widehat{\mathbf{f}}$. Such a tool is provided by Theorem 2: it says that the value of β should be increased.

B. Algorithm

Given $\beta > 0$, the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}$ serves as auxiliary information for sorting the pixels of $\mathbf{u}$ in a strict way. More precisely, 2-tuples $(\mathbf{u}[i], \widehat{\mathbf{f}}[i])$ are used as described next[4].

$$i \neq j \quad \begin{cases} \Big[\ \mathbf{u}[i] < \mathbf{u}[j]\ \Big] \ \text{ or } \ \Big[\ \mathbf{u}[i] = \mathbf{u}[j] \ \text{ and } \ \widehat{\mathbf{f}}[i] < \widehat{\mathbf{f}}[j]\ \Big] & \Rightarrow \quad i \prec j \ ; \\[2ex] \Big[\ \mathbf{u}[i] > \mathbf{u}[j]\ \Big] \ \text{ or } \ \Big[\ \mathbf{u}[i] = \mathbf{u}[j] \ \text{ and } \ \widehat{\mathbf{f}}[i] > \widehat{\mathbf{f}}[j]\ \Big] & \Rightarrow \quad i \succ j \ . \end{cases} \tag{14}$$

If it occurs that for some $i \neq j$ we have $\mathbf{u}[i] = \mathbf{u}[j]$ and $\widehat{\mathbf{f}}[i] = \widehat{\mathbf{f}}[j]$, then the current parameter β is increased according to the rule

$$\beta \leftarrow c\beta \quad \text{for} \quad c > 1 \ .$$

Since $\widehat{\mathbf{f}}$ is a restored version of $\mathbf{u}$ (hence "better" than $\mathbf{u}$), it is worth considering the substitution

$$\mathbf{u} \leftarrow \widehat{\mathbf{f}} \ .$$

Then the minimizer of the new $\mathcal{J}$ is computed. If necessary, this update is repeated until (14) can be applied to *all* pairs $(i, j) \in \mathbb{I}_n \times \mathbb{I}_n$, $i \neq j$. Let us anticipate that according to our experiments, this procedure needs to be done more than once only in some special cases.

The resulting algorithm for exact histogram specification using the variational approach presented in section III is summarized in Algorithm 1.

Algorithm 1 Exact Histogram Specification Using Variational Approach

---

Input: the input image $\mathbf{u}$, the specified histogram $\mathbf{h}$ and the intensity value set $\mathbf{p}$.
Output: the specified image $\mathbf{v}$.

1: Initialize α_1, α_2 and β, c.
2: Set the tuple κ as $\kappa[i] = \mathbf{u}[i]$, $\forall i \in \mathbb{I}_n$.
3: **while** There are equal-valued entries in κ **do**
4: $\quad \widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \mathcal{J}(\mathbf{f}, \mathbf{u})$;
5: $\quad$ Update the tuple $\kappa[i] \leftarrow \big(\kappa[i], \widehat{\mathbf{f}}[i]\big)$, $\forall i \in \mathbb{I}_n$;
6: $\quad$ Update $\mathbf{u} \leftarrow \widehat{\mathbf{f}}$ and $\beta \leftarrow c\beta$;
7: **end while**
8: Start from the first pixels on the ordering list, assign the first $\mathbf{h}[1]$ pixels with intensity value $\mathbf{p}[1]$, the next $\mathbf{h}[2]$ pixels with intensity value $\mathbf{p}[2]$, and so on until all pixels are assigned to their new intensities; the resulting image is the specified image $\mathbf{v}$.
9: **return** $\mathbf{v}$.

---

[4]In general, if $\widehat{\mathbf{f}}[i] < \widehat{\mathbf{f}}[j]$, we can not deduce that $\mathbf{u}[i] \leq \mathbf{u}[j]$ because $\widehat{\mathbf{f}}$ may not preserve the natural order of the pixel values of the input digital image $\mathbf{u}$.

Steps 3-7 in the algorithm are clearly designed to reach a solution $\widehat{\mathbf{f}}$ such that (14) can be applied to all pairs (i, j), $i \neq j$. This algorithm is quite general and can be applied using any functions (ψ, ϕ) obeying H1 and Q1—for some examples see Table I.

C. Implementation of Algorithm 1

We apply the Algorithm 1 with

$$\psi(t) = \sqrt{\alpha_1 + t^2} \quad \text{and} \quad \phi(t) = \sqrt{\alpha_2 + t^2} \quad \text{for} \quad \alpha_1 = \alpha_2 \tag{15}$$

in (2). Based on Fig. 3 and the discussion that precede it, we consider that Φ in (4) is defined using *the four adjacent neighbors*.

There are many algorithms in literature to compute the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}(\cdot, \mathbf{u})$ in step 4 of the Algorithm. These include partial differential equation based methods such as explicit method [47], semi-implicit method [33], operator splitting [36], lagged diffusivity fixed point iterations [49], Polak-Ribière CG method [5], majorization-minimization algorithms [6], [22], [28], Nesterov algorithms [35] and so on.

We apply lagged diffusivity fixed point iterations to solve the Euler-Lagrange equation derived from (2), i.e. to solve step 4 in the Algorithm. The data-fitting term and the regularization term are linearized by a quadratic formulation in the each iteration. Given our choice in (15), the intermediate solution $\mathbf{f}_k$ in k-th iteration is the minimizer of the following function

$$\|\mathbf{f} - \mathbf{u}\|^2_{\boldsymbol{D}_{1,k}} + \beta \|G\mathbf{f}\|^2_{\boldsymbol{D}_{2,k}}.$$

Here $\boldsymbol{D}_{1,k}$ and $\boldsymbol{D}_{2,k}$ are the diagonal matrix with the diagonal element $1/\sqrt{\alpha_1 + \mathbf{f}_{k-1}[i]}$ and $1/\sqrt{\alpha_2 + \mathbf{g}_i^T \mathbf{f}_{k-1}}$ respectively. Therefore, a linear system is obtained in each iteration. We apply conjugate gradient method to solve the corresponding linear system.

Given the choice for (ψ, ϕ) in (15) and the selected minimization method, we could fix the parameter c in step 6 of Algorithm 1 to

$$c = 2 \ .$$

Remark 3. *It may be curious to note that for* (ψ, ϕ) *in* (15)*, all experiments have shown that for* $\alpha_1 = \alpha_2 \leq 0.05$ *and* $\beta \leq 0.2$ *(the typical values for the parameters used in our experiments, we always had* $\|\widehat{\mathbf{f}} - \mathbf{u}\|_\infty < 0.35$*. In any case, if* $\|\widehat{\mathbf{f}} - \mathbf{u}\|_\infty < 0.5$*, reminding that the set* $\mathcal{P}$ *in* (1) *is composed of integers, it is obvious that* $\widehat{\mathbf{f}}$ *preserves* [5] *the order of the pixel values of* $\mathbf{u}$*. Then we can assign all pixels to the new intensities given by the prescribed histogram by using only* $\widehat{\mathbf{f}}$ *which leads to the sought-after* $\mathbf{v}$.

V. Experimental Results

In this section, two practical problems: contrast compression and equalization inversion, were tested to show the performance of the proposed method for exact histogram specification. We compare our method with the local mean (LM) algorithm [16] for $K = 6$ and with the wavelet-based algorithm (WA) [50] for Haar wavelet as recommended by the authors. The experiments were performed under Mac OS X 10.7.2 and MATLAB v7.12 on a MacBook Air Laptop with an Intel Core i5 1.7 GHZ processor and 4GB of RAM. For our method, we set α_i, $i = 1, 2$, to 0.05 and β to 0.1 (see in (2)) and $c = 2$ in Algorithm 1. We applied fixed point method to compute the minimizer $\widehat{\mathbf{f}}$ of $\mathcal{J}(\cdot, \mathbf{u})$ in (2). In each iteration of the fixed point method, a

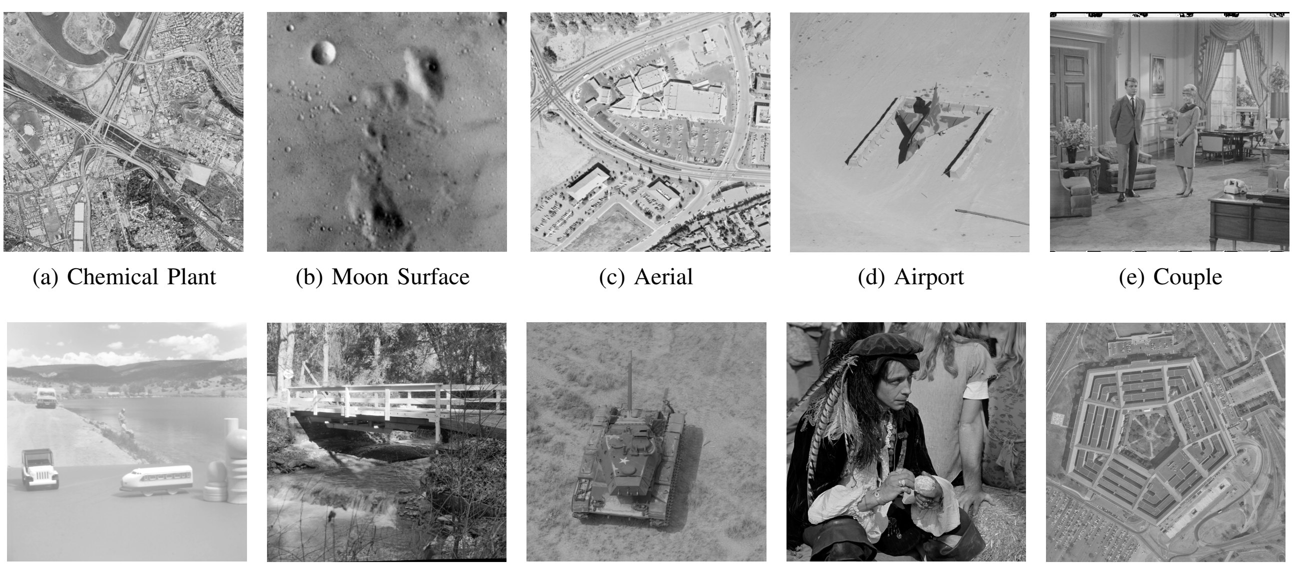
[5] It is easy to establish that $\widehat{\mathbf{f}}[i] > \widehat{\mathbf{f}}[j]$ entails $\mathbf{u}[i] \geq \mathbf{u}[j]$ and vice-versa, that $\widehat{\mathbf{f}}[i] < \widehat{\mathbf{f}}[j]$ entails $\mathbf{u}[i] \leq \mathbf{u}[j]$.

linear system is obtained and solved by conjugate gradient (CG) method. In our numerical tests, we stop the iteration of the fixed point method when the relative difference between the iterant is less than 10^{-6} or the number of iteration reaches to 10. The stopping criterion of CG is that the relative difference between the iterant is less than 10^{-6} and the number of the iteration reaches to 50. In both applications, we have realized a large number of experiment (50 at least). In all cases, we obtained results similar to those presented next.

In this section, we describe the experiments done with 10 true quantized images, shown in Figure 4, available at `http://sipi.usc.edu/database/`. Their sizes are 256×256 for images (a)–(b), 512×512 for images (c)–(h), and 1024×1024 for images (i)–(j). In order to measure the results quantitatively, we start out with a given true quantized image $\mathbf{w}$ with histogram $\mathbf{h}_{\mathbf{w}}$, we degrade it to obtain an input quantized image $\mathbf{u}$. By applying the three methods on $\mathbf{u}$ with prescribed histogram $\mathbf{h}_{\mathbf{w}}$, we obtain an output image $\mathbf{v}$ which is in fact a restored version of $\mathbf{w}$. We use peak-signal-to-noise-ratio to measure the quality of the output image $\mathbf{v}$ with respect to $\mathbf{w}$. It is defined as $\mathrm{PSNR} = 20\log_{10}(255NM/\|\mathbf{v}-\mathbf{w}\|_2)$.

A. *Restoration of Contrast Compression*

The input image $\mathbf{u}$ is obtained from $\mathbf{w}$ by the degradation: $\mathbf{u} = \mathrm{round}(\rho\cdot\mathbf{w})$, where $\rho < 1$ is a constant. We show the process of contrast compression in Figure 5. This situation arises when a picture is taken with insufficient exposure time, or when we want to compress the image by reducing the number of intensity levels. For example, a 7-bit image can be obtained from an 8-bit image by using $\rho = 0.5$. In the tests, we used LM, WA and our method to obtain the output images $\mathbf{v}$ having a prescribed histogram $\mathbf{h}_{\mathbf{w}}$. The comparisons of LM, WA and our algorithm are shown in Table II. We see from the PSNR values that our method outperforms LM and WA in all cases. It demonstrates that our algorithm yields the best restoration. We also show the number of the updates of β and CPU time (second) to obtain the strictly order for our method, see Table III.



(a) Chemical Plant (b) Moon Surface (c) Aerial (d) Airport (e) Couple

(f) Motion Car (g) Stream and Bridge (h) Tank (i) Man (j) Pentagon

Fig. 4: The input quantized images. The sizes of the images are 256×256 for images (a)–(b), 512×512 for images (c)–(h), and 1024×1024 for images (i)–(j), respectively.

One important indicator for a good exact histogram specification algorithm is to see if it can establish a strict ordering for all the pixels. If a sorting method yields two pixels sharing the same value we call them a equal-valued pixel, and consider that as a failure of the method. Table IV shows the numbers of equal-valued

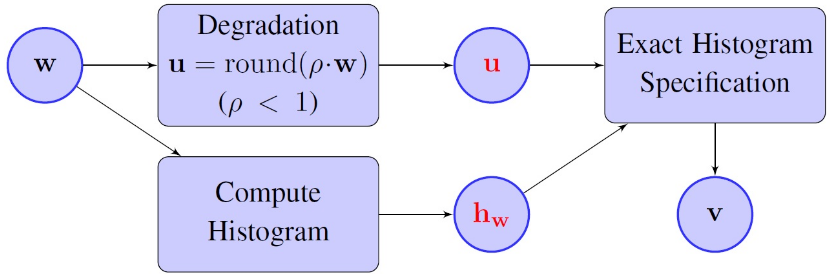


Fig. 5: Contrast Compression.

pixels produced by the three methods. We find that LM and WA have a high number of equal-valued pixels while our method can give a total ordering of all pixels for all images.

	$\rho = 0.5$			$\rho = 0.25$			$\rho = 0.125$		
Image	LM	WA	Ours	LM	WA	Ours	LM	WA	Ours
Chemical Plant	51.45	51.36	51.42	44.89	44.64	44.89	39.42	38.84	39.45
Moon Surf	51.47	51.44	51.45	44.92	44.77	44.92	39.57	39.27	39.61
Aerial	51.61	51.49	51.66	45.15	44.81	45.29	39.68	39.12	39.85
Airport	67.01	66.87	67.13	48.70	48.46	48.77	42.19	41.92	42.51
Couple	51.72	51.59	51.77	45.36	45.00	45.57	40.28	39.61	40.61
Motion Car	53.42	53.01	53.50	48.29	47.42	48.75	42.75	41.80	43.56
Stream and Bridge	∞	∞	∞	∞	∞	∞	39.97	39.73	40.04
Tank	58.70	58.60	58.74	46.01	45.78	46.00	40.19	39.73	40.15
Man	51.70	51.61	51.70	45.13	44.85	45.13	40.35	39.77	40.39
Pentagon	51.47	51.45	51.49	44.93	44.84	45.02	39.50	39.28	39.71

TABLE II: The PSNR (dB) between the true image $\mathbf{w}$ and the output image $\mathbf{v}$. Here "∞" denote that the output image $\mathbf{v}$ is exactly the same with the true image $\mathbf{w}$.

	$\rho = 0.5$		$\rho = 0.25$		$\rho = 0.125$	
Image	# of β	CPU Time	# of β	CPU Time	# of β	CPU Time
Chemical Plant	1	0.45	1	0.44	1	0.43
Moon Surf	1	0.42	1	0.43	1	0.48
Aerial	1	1.87	1	1.85	1	2.04
Airport	1	1.91	1	1.57	2	5.25
Couple	1	1.96	1	1.84	1	1.69
Motion Car	1	1.69	2	5.28	3	20.59
Stream and Bridge	1	1.98	1	2.07	1	2.06
Tank	1	2.08	1	1.89	1	1.82
Man	1	7.13	1	6.49	2	16.17
Pentagon	1	6.74	1	7.29	1	6.69

TABLE III: The number of updates of β and CPU time (second) to obtain the strictly order from our method.

B. Histogram Equalization Inversion

We will consider the application of the histogram equalization inversion, which is to recover the true quantized image $\mathbf{w}$ with histogram $\mathbf{h}_w$ from its specified version $\mathbf{u}$. Let $\mathbf{u} = T(\mathbf{w}, \mathbf{h})$ be the process to specify the image $\mathbf{w}$ with histogram $\mathbf{h}_w$ to an image $\mathbf{u}$ such that the histogram of $\mathbf{u}$ is $\mathbf{h}$. The quantized

	$\rho = 0.5$			$\rho = 0.25$			$\rho = 0.125$		
Image	LM	WA	Ours	LM	WA	Ours	LM	WA	Ours
Chemical Plant	1	18	0	2	41	0	36	252	0
Moon Surf	0	0	0	0	15	0	4	309	0
Aerial	0	190	0	323	2,203	0	3,084	9,411	0
Airport	18	1579	0	2,175	12,436	0	31,638	63,875	0
Couple	59	378	0	321	2,100	0	2,747	12,947	0
Motion Car	3,714	16,650	0	34,027	54,935	0	78,701	101,468	0
Stream and Bridge	576	1,072	0	577	1,077	0	921	2849	0
Tank	0	15	0	0	198	0	281	2,176	0
Man	81	1,290	0	879	6,162	0	7,577	28,076	0
Pentagon	0	14	0	0	344	0	33	5,167	0

TABLE IV: The numbers of equal-valued pixels from the three methods.

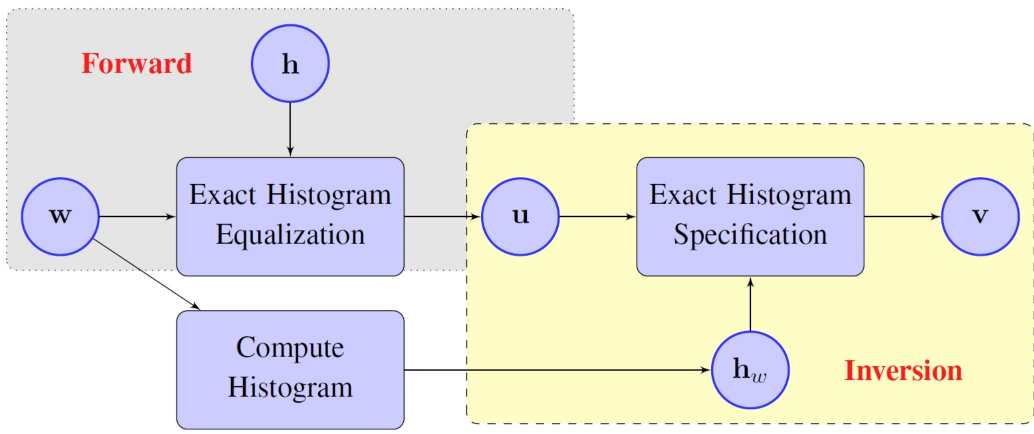


Fig. 6: Histogram Equalization Inversion. A true quantized image $\mathbf{w}$ with histogram $\mathbf{h}_w$ is equalized to the input image $\mathbf{u}$ using exact histogram specification method. The task is to restore $\mathbf{w}$ from the input image $\mathbf{u}$ and the prescribed histogram $\mathbf{h}_w$ using the same exact histogram specification method. The specified image $\mathbf{v}$ is a restored version of $\mathbf{w}$.

image $\mathbf{w}$ can be exactly recovered by $T(\mathbf{u}, \mathbf{h}_w)$ under the hypothesis of order preservation by the method. Since the ordering among the pixels of $\mathbf{w}$ is not identical with that among pixels of $\mathbf{u}$, the recovered image $\mathbf{v} = T(\mathbf{u}, \mathbf{h}_w)$ is just an approximation of $\mathbf{w}$. We show the process of histogram equalization inversion in Figure 6.

Table V shows the PSNR of the results by the three methods. We notice from Table V that WA method yields better PSNR than LM method in all images, but worse than our method in all cases except for the "Man" images. The number of the update of β and CPU time (second) to obtain the strictly order for our method is shown in Table VI.

Figures 8–11 give the enlarged portions of the difference images on "Aerial", "Couple", "Motion Car", "Man" images, the corresponding enlarged portions of the input image are shown in Figure 7. We can discern more features in the difference images by both LM method and WA method than by our method. Though WA method yields better PSNR than our method on "Man" image, from Figure 11, we can also observe that our method yields the fewest features in the difference image. It demonstrates that our algorithm yields the best restoration.

We also compare the failure of three methods. We denote the procedure to get the pixel ordering of $\mathbf{w}$ as

Image	LM	WA	Ours	Image	LM	WA	Ours
Chemical Plant	49.34	49.64	49.67	Motion Car	54.40	55.01	54.80
Moon Surf	47.39	47.46	47.85	Stream and Bridge	44.76	45.00	45.08
Aerial	48.36	48.59	50.08	Tank	48.11	48.25	48.35
Airport	46.74	46.81	47.24	Man	49.23	49.48	49.45
Couple	44.22	44.33	49.18	Pentagon	50.69	50.86	51.36

TABLE V: The PSNR (dB) between the true image $\mathbf{w}$ and output images $\mathbf{v}$.

Fig. 7: Enlarged portions of "Aerial" image, "Couple" image, "Motion Car" image and "Man" image, respectively.

"Forword" and the procedure to get the pixel ordering of $\mathbf{u}$ as "Backword". Table VII shows the numbers of equal-valued pixels produced by the three methods. We find that LM and WA have a high number of equal-valued pixels while our method can give a total ordering of all pixels for all three images.

	Forward		Backward	
Image	# of β	CPU Time	# of β	CPU Time
Chemical Plant	1	0.40	1	0.48
Moon Surf	1	0.54	1	0.49
Aerial	1	1.95	1	2.06
Airport	1	2.09	1	2.06
Couple	1	1.88	1	1.85
Motion Car	1	1.92	1	2.02
Stream and Bridge	1	1.90	1	1.86
Tank	1	1.98	1	2.02
Man	1	7.49	1	7.76
Pentagon	1	7.54	1	7.75

TABLE VI: The number of updates of β and CPU time (second) to obtain the strictly order from our method.

VI. Conclusions and perspectives

In this paper, we propose a variational approach for exact histogram specification. Since the energy we minimize is smooth, its minimizers enable us to strictly order all the pixels in the image. Noticing also that our method reduces the quantification noise, the obtained results outperform the preexisting methods.

VII. Appendix

Given a square matrix A, the expression $A \succ 0$ means that A is positive definite and $A \succeq 0$ that A is positive semi-definite.

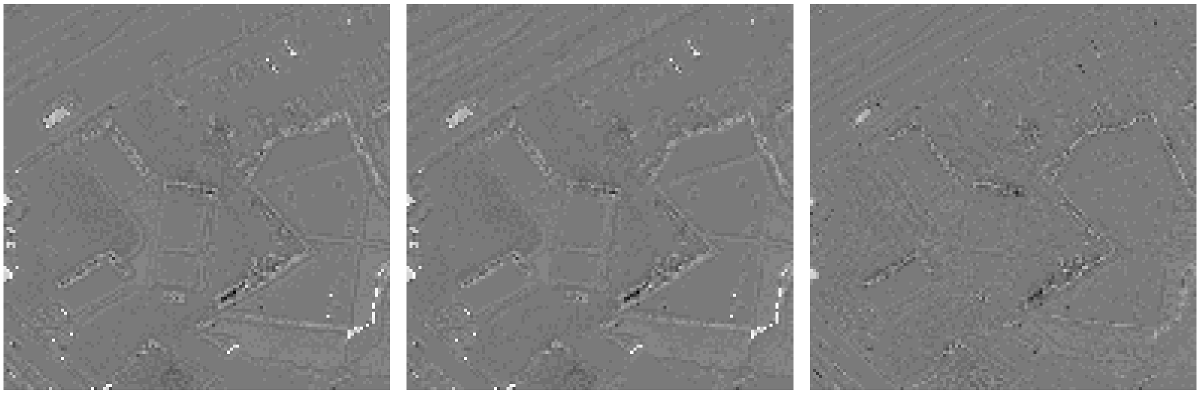


Fig. 8: The enlarged portions of the different images between $\mathbf{w}$ and $\mathbf{v}$ by LM (Left), WA (Middle) and our method (Right) for "Aerial" image. Our method yields fewest features in the difference images.

Fig. 9: The enlarged portions of the different images between $\mathbf{w}$ and $\mathbf{v}$ by LM (Left), WA (Middle) and our method (Right) for "Couple" image. Our method yields fewest features in the difference images.

A. *Proof of Proposition 1*

By H1, for any $\mathbf{u} \in \mathbb{R}^n$, the function $\mathcal{J}(\cdot, \mathbf{u})$ in (2) is strictly convex and coercive, hence for any $\mathbf{u}$ and $\beta > 0$, it has a unique minimizer. Each minimizer point $\widehat{\mathbf{f}}$ of $\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u})$ is determined by $D_1\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u}) = 0$. We have

$$0 = D_1\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u}) = D_1\Psi(\widehat{\mathbf{f}}, \mathbf{u}) + \beta D_1\Phi(\widehat{\mathbf{f}}), \tag{16}$$

where

$$D_1\Psi(\widehat{\mathbf{f}}, \mathbf{u}) = \begin{bmatrix} \psi'(\widehat{\mathbf{f}}[1] - \mathbf{u}[1]) \\ \cdots \\ \psi'(\widehat{\mathbf{f}}[p] - \mathbf{u}[p]) \end{bmatrix}^T \quad \text{and} \quad D_1\Phi(\widehat{\mathbf{f}}) = \begin{bmatrix} \phi'(\mathbf{g}_1^T\widehat{\mathbf{f}}) \\ \cdots \\ \phi'(\mathbf{g}_r^T\widehat{\mathbf{f}}) \end{bmatrix}^T G \ . \tag{17}$$

Differentiation with respect to $\widehat{\mathbf{f}}$ yet again yields

$$D_1^2\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u}) = D_1^2\Psi(\widehat{\mathbf{f}}, \mathbf{u}) + \beta D_1^2\Phi(\widehat{\mathbf{f}}) \in \mathbb{R}^{n\times n}. \tag{18}$$

Here, $D_1^2\Psi(\widehat{\mathbf{f}}, \mathbf{u})$ is an $n \times n$ diagonal matrix with strictly positive entries according to H1:

$$D_1^2\Psi(\widehat{\mathbf{f}}, \mathbf{u})[i,i] = \psi''(\widehat{\mathbf{f}}[i] - \mathbf{u}[i]), \quad \forall i \in \mathbb{I}_n \ . \tag{19}$$

Hence $D_1^2\Psi(\mathcal{F}(\mathbf{u}), \mathbf{u}) \succ 0$. Furthermore,

$$D_1^2\Phi(\widehat{\mathbf{f}}) = G^T \mathrm{diag}\Big(\phi''(\mathbf{g}_1^T\widehat{\mathbf{f}}), \cdots, \phi''(\mathbf{g}_r^T\widehat{\mathbf{f}})\Big)G, \tag{20}$$

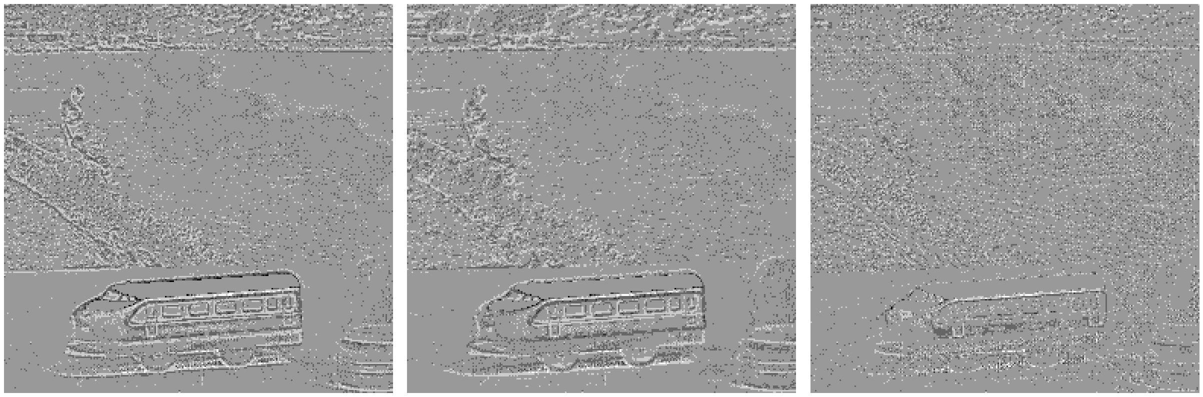


Fig. 10: The enlarged portions of the different images between $\mathbf{w}$ and $\mathbf{v}$ by LM (Left), WA (Middle) and our method (Right) for "Tank" image. Our method yields fewest features in the difference images.

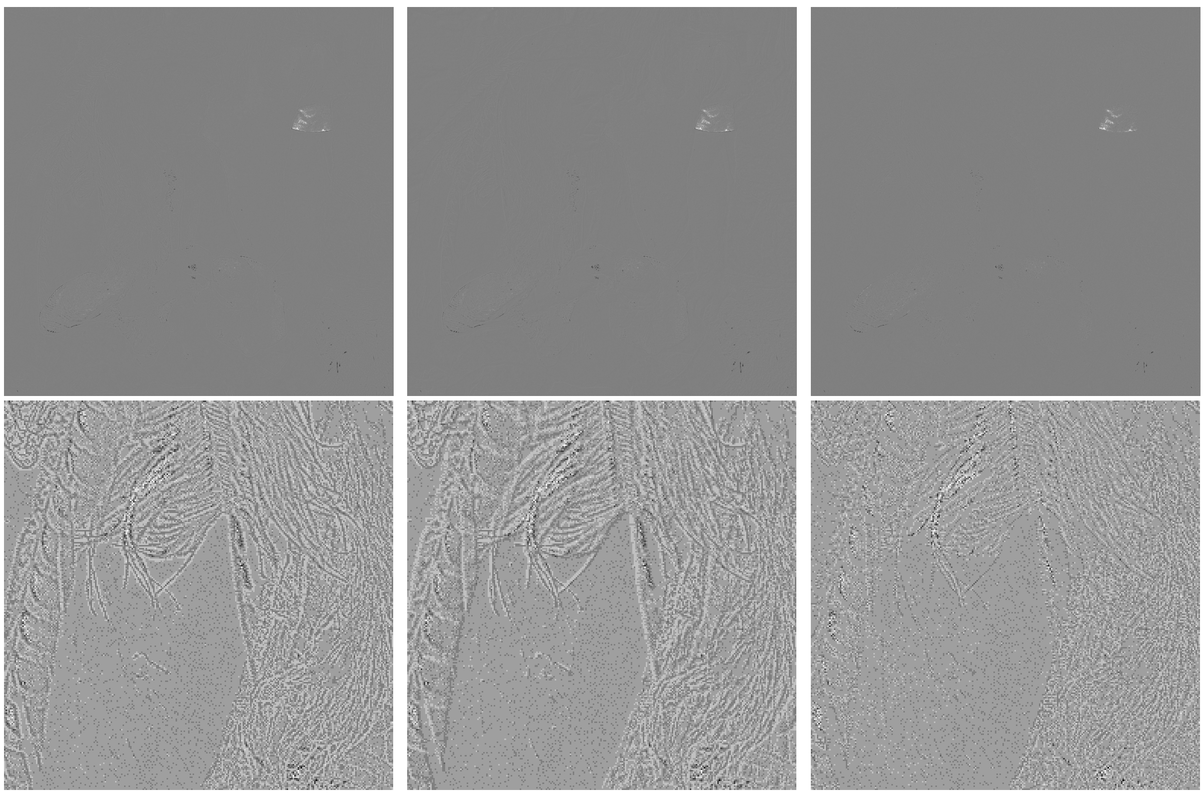


Fig. 11: The difference images of Man. Top: The difference imageS between $\mathbf{w}$ and $\mathbf{v}$ by LM (Left), WA (Middle) and our method (Right) for motion car image. Bottom: the enlarged portions of the different images. Our method yields fewest features in the difference images.

so $D_1^2\Phi(\widehat{\mathbf{f}}) \succeq 0$. It follows that $D_1^2\mathcal{J}(\widehat{\mathbf{f}}, \mathbf{u}) \succ 0$, for any $\mathbf{u} \in \mathbb{R}^n$.

Consequently, Lemma 1 holds true for any $\mathbf{u} \in \mathbb{R}^n$ and any $\beta > 0$. The same lemma shows that the statement of Proposition 1 holds true for $\mathcal{O} = \mathbb{R}^n$.

	Forward			Backword		
Image	LM	WA	Ours	LM	WA	Ours
Chemical Plant	1	18	0	2	2	0
Moon Surf	0	0	0	36	34	0
Aerial	0	0	0	0	0	0
Airport	19	1751	0	52	52	0
Couple	48	201	0	11	15	0
Motion Car	45	1,968	0	388	388	0
Stream and Bridge	577	1,071	0	371	392	0
Tank	0	8	0	28	29	0
Man	58	907	0	31	32	0
Pentagon	0	1	0	42	42	0

TABLE VII: The numbers of equal-valued pixels from the three methods.

B. *Proof of Lemma 2*

Since $\mathcal{F}$ is a local minimizer function,

$$D_1\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u}) = 0, \quad \forall \mathbf{u}\in\mathbb{R}^n. \tag{21}$$

We can thus differentiate with respect to $\mathbf{u}$ on both sides of (21) and yields

$$D_1^2\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u})D\mathcal{F}(\mathbf{u}) + D_2D_1\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u}) = 0, \quad \forall \mathbf{u}\in\mathbb{R}^n. \tag{22}$$

Note that $D\mathcal{F}(\mathbf{u})$ and $D_2D_1\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u})$ are $n\times n$ real matrices. The Hessian matrix $H(\mathbf{u}) = D_1^2\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u})$ can be expanded using (18):

$$H(\mathbf{u}) = D_1^2\Psi(\mathcal{F}(\mathbf{u}),\mathbf{u}) + \beta D_1^2\Phi(\mathcal{F}(\mathbf{u})) \in \mathbb{R}^{n\times n}.$$

Replacing $\widehat{\mathbf{f}}$ by $\mathcal{F}(\mathbf{u})$ in (19) and (20) yields

$$H(\mathbf{u}) = \mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u}) - \mathbf{u}[i])\}_{i=1}^n\big) + \beta G^T\mathrm{diag}\big(\{\phi''(\mathbf{g}_i^T\mathcal{F}(\mathbf{u}))\}_{i=1}^n\big)G\ ,$$

as stated in (6). Using H1, it is readily seen that

$$\mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u}) - \mathbf{u}[i])\}_{i=1}^n\big) \succ 0 \quad \text{and} \quad G^T\mathrm{diag}\big(\{\phi''(\mathbf{g}_i^T\mathcal{F}(\mathbf{u}))\}_{i=1}^n\big)G \succeq 0.$$

Then $H(\mathbf{u}) \succ 0$, hence $H(\mathbf{u})$ is invertible.

Using (16) and (17), where we consider $\mathcal{F}(\mathbf{u})$ in place of $\widehat{\mathbf{f}}$, shows that

$$D_2D_1\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u}) = D_2D_1\Psi(\mathcal{F}(\mathbf{u}),\mathbf{u}) = -\mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u}) - \mathbf{u}[i])\}_{i=1}^n\big)\ . \tag{23}$$

Then (22) is equivalent to

$$D\mathcal{F}(\mathbf{u}) = -\Big(H(\mathbf{u})\Big)^{-1}D_2D_1\mathcal{J}(\mathcal{F}(\mathbf{u}),\mathbf{u}) = \Big(H(\mathbf{u})\Big)^{-1}\mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u}) - \mathbf{u}[i])\}_{i=1}^n\big), \quad \forall \mathbf{u}\in\mathbb{R}^n\ . \tag{24}$$

Obviously, $\mathrm{rank}\big(D\mathcal{F}(\mathbf{u})\big) = n$. This result is independent of the value of $\beta > 0$.

C. Proof of Theorem 1

Given $g \in \mathcal{G}$, where $\mathcal{G}$ is given in (8), consider the function $f_g : \mathbb{R}^n \to \mathbb{R}$ defined by

$$f_g(\mathbf{u}) = g^T \mathcal{F}(\mathbf{u}), \quad \forall \mathbf{u} \in \mathbb{R}^n \ ,$$

as well as the inverse image of the origin

$$K_g \stackrel{\text{def}}{=} f_g^{-1}(0) = \{\mathbf{u} \in \mathbb{R}^n \ \mid \ g^T \mathcal{F}(\mathbf{u}) = 0\} \ . \tag{25}$$

Using Proposition 1, f_g is $\mathcal{C}^{s-1}$-continuous, so K_g is closed. By Lemma 2, $D\mathcal{F}(\mathbf{u})$ is invertible. Hence $g^T D\mathcal{F}(\mathbf{u}) \neq 0$ and thus

$$\operatorname{rank}\big(f_g(\mathbf{u})\big) = \operatorname{rank}\big(g^T D\mathcal{F}(\mathbf{u})\big) = 1 \ .$$

By an extension[6] of the constant rank theorem [4], the subset K_g in (25), supposed nonempty[7], is a $\mathcal{C}^{s-1}$ manifold of $\mathbb{R}^n$ of dimension $n-1$. Hence $\mathbb{L}^n(K_g) = 0$ (see e.g. [20], [32]. The set $K_{\mathcal{G}}$ in (9) also reads

$$K_{\mathcal{G}} = \bigcup_{g \in \mathcal{G}} K_g.$$

Using that $\mathcal{G}$ is of finite cardinality, it follows that $K_{\mathcal{G}}$ is closed in $\mathbb{R}^n$ and that

$$\mathbb{L}^n\left(K_{\mathcal{G}}\right) = 0.$$

The conclusion is clearly independent of the value of $\beta > 0$.

D. Proof of Proposition 2

Given $(i,j) \in \mathbb{I}_n \times \mathbb{I}_n$ (including $i = j$), define the subset $K_{i,j} \subset \mathbb{R}^n$ as

$$K_{i,j} = \mathcal{F}_i^{-1}(\mathbf{u}_j) = \big\{\mathbf{u} \in \mathbb{R}^n \ \mid \ \mathcal{F}_i(\mathbf{u}) = \mathbf{u}_j\big\} \ . \tag{26}$$

For some $p \in \mathcal{P}$ and $i \in \mathbb{I}_n$, put

$$K_{p,i} = \mathcal{F}_i^{-1}(p) = \big\{\mathbf{u} \in \mathbb{R}^n \ \mid \ \mathcal{F}_i(\mathbf{u}) = p\big\} \ . \tag{27}$$

From Proposition 1, $\mathcal{F}_i : \mathbb{R}^n \to \mathbb{R}$ is $\mathcal{C}^{s-1}$ continuous, so $K_{i,j}$ and $K_{p,i}$ are closed in $\mathbb{R}^n$. By Lemma 2, all rows of $D\mathcal{F}(\mathbf{u}) \in \mathbb{R}^{n \times n}$ are linearly independent, $\forall \mathbf{u} \in \mathbb{R}^n$. Consequently, for any $i \in \mathbb{I}_n$,

$$\operatorname{rank}\big(D\mathcal{F}_i(\mathbf{u})\big) = 1, \quad \forall \mathbf{u} \in \mathbb{R}^n \ .$$

[6]We use the following extension of the constant rank theorem, restated in our context (for details one can check [4, p. 96]). Let f be a $\mathcal{C}^s$ application from an open set $\mathcal{O} \subset \mathbb{R}^n$ to $\mathbb{R}$. Assume that $Df(\mathbf{u})$ has constant rank r for all $\mathbf{u} \in \mathcal{O}$. Given a $c \in \mathbb{R}$, the inverse image $f^{-1}(c)$ (supposed nonempty) is a $\mathcal{C}^s$-manifold of $\mathbb{R}^n$ of dimension $n - r$.

[7]Even though $\operatorname{rank}\big(g^T D\mathcal{F}(\mathbf{u})\big) = 1$, we can have $K_g \neq \varnothing$. For instance, let

$$\mathcal{F}(\mathbf{u}) = \begin{bmatrix} \mathbf{u}[1] \\ \mathbf{u}[2] \end{bmatrix} .$$

Obviously,

$$D\mathcal{F}(\mathbf{u}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \operatorname{rank}\big(D\mathcal{F}(\mathbf{u})\big) = 2.$$

Then

$$K_{\mathcal{G}} = \{\mathbf{u} \in \mathbb{R}^2 \ \mid \ \mathbf{u}[1] = \mathbf{u}[2]\}, \quad \mathbb{L}^2(K_{\mathcal{G}}) = 0 \ .$$

Indeed, $K_{\mathcal{G}}$ is a one-dimensional subspace in $\mathbb{R}^2$, hence it is closed and of measure zero.

Using the same extension of the constant rank theorem as in the proof of Theorem 1 [4], $K_{i,j}$ and $K_{p,i}$ are $\mathcal{C}^{s-1}$ submanifolds of $\mathbb{R}^n$ of dimension $n-1$. Then [20], [32]

$$\mathbb{L}^n(K_{i,j}) = 0 \quad \text{and} \quad \mathbb{L}^n(K_{p,i}) = 0.$$

Noticing that $K_{\mathcal{I}}$ in (11) and $K_{\mathcal{P}}$ in statement (ii) are finite unions of $(n-1)$-dimensional submanifolds in $\mathbb{R}^n$ like $K_{i,j}$ and $K_{p,i}$, respectively, entails the result. The independence of these results from $\beta > 0$ is obvious.

E. A Lemma needed to prove Proposition 3

Lamma 3. *Let* $(A,\, B) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n}$ *satisfy*

$$\begin{array}{rcl} A & = & \mathrm{diag}\big(\{a_i\}_{i=1}^n\big) \;\; \textit{where} \;\; a_i > 0, \;\; \forall i \in \mathbb{I}_n \;, \\ B & = & B^T, \;\; B \succeq 0 \;. \end{array}$$

Consider the $n \times n$ *matrix* $M \overset{\text{def}}{=} (A+B)^{-1}A$. *Then all eigenvalues of* M^TM *belong to* $(0, 1]$.

Proof: Let λ be a (right) eigenvalue of $(A+B)^{-1}A$ and $\mathbf{v} \in \mathbb{R}^n \setminus \{0\}$ an eigenvector corresponding to λ. Then

$$\begin{array}{rrcl} & \lambda\mathbf{v} & = & M\mathbf{v} \\ \Leftrightarrow & \lambda\mathbf{v} & = & (A+B)^{-1}A\mathbf{v} \qquad (28) \\ \Leftrightarrow & \lambda(A+B)\mathbf{v} & = & A\mathbf{v} \\ \Leftrightarrow & \lambda B\mathbf{v} & = & (1-\lambda)A\mathbf{v} \;. \qquad (29) \end{array}$$

If $\lambda = 0$ then (29) yields $A\mathbf{v} = 0$ which is impossible because $A \succ 0$, $A^T = A$ and $\mathbf{v} \neq 0$. Hence

$$\lambda \neq 0 \;. \tag{30}$$

Furthermore, (29) yields

$$\lambda\, \mathbf{v}^T B\mathbf{v} = (1-\lambda)\, \mathbf{v}^T A\mathbf{v} \;.$$

Using that $A \succ 0$ and $B \succeq 0$, the last equation shows that

$$\frac{1}{\lambda} - 1 = \frac{\mathbf{v}^T B\mathbf{v}}{\mathbf{v}^T A\mathbf{v}} \geq 0 \;. \tag{31}$$

Combining the latter inequality with (30) entails that

$$0 < \lambda \leq 1 \;. \tag{32}$$

Hence all eigenvalues of M live in $(0, 1]$.

Using that A is diagonal with positive diagonal entries, we can write down

$$A + B = A^{\frac{1}{2}}\left(I + A^{-\frac{1}{2}}BA^{-\frac{1}{2}}\right)A^{\frac{1}{2}}. \tag{33}$$

Then the definition of M shows that[8]

$$\begin{array}{rcl} M^TM & = & A(A+B)^{-2}A \\ & = & \left(I + A^{-\frac{1}{2}}BA^{-\frac{1}{2}}\right)^{-2} \;. \qquad (34) \end{array}$$

[8]Remind that $A = A^T$.

Using (33), the expression in (28) is *equivalent* to

$$\begin{array}{rrcl} & \lambda\mathbf{v} & = & A^{-\frac{1}{2}}(I+A^{-\frac{1}{2}}BA^{-\frac{1}{2}})^{-1}A^{-\frac{1}{2}}A\mathbf{v} \\ \Leftrightarrow & \lambda\left(A^{\frac{1}{2}}\mathbf{v}\right) & = & (I+A^{-\frac{1}{2}}BA^{-\frac{1}{2}})^{-1}\left(A^{\frac{1}{2}}\mathbf{v}\right) \end{array}$$

Combining the last result with (34) yields

$$\begin{array}{rcl} M^TM\left(A^{\frac{1}{2}}\mathbf{v}\right) & = & \left(I+A^{-\frac{1}{2}}BA^{-\frac{1}{2}}\right)^{-1}\left(I+A^{-\frac{1}{2}}BA^{-\frac{1}{2}}\right)^{-1}\left(A^{\frac{1}{2}}\mathbf{v}\right) \\ & = & \lambda\left(I+A^{-\frac{1}{2}}BA^{-\frac{1}{2}}\right)^{-1}\left(A^{\frac{1}{2}}\mathbf{v}\right) \\ & = & \lambda^2\left(A^{\frac{1}{2}}\mathbf{v}\right) . \end{array}$$

Consequently, λ^2 is an eigenvalue of M^TM corresponding to an eigenvector given by $\left(A^{\frac{1}{2}}\mathbf{v}\right)$. Thus all eigenvalues of M^TM belong to $(0,1]$ as well.

F. *Proof of Proposition 3*

Let us denote

$$\begin{array}{rcl} A(\mathbf{u}) & = & \mathrm{diag}\big(\{\psi''(\mathcal{F}_i(\mathbf{u})-\mathbf{u}[i])\}_{i=1}^n\big) , \\ B(\mathbf{u}) & = & G^T\mathrm{diag}\big(\{\phi''(\mathbf{g}_i^T\mathcal{F}(\mathbf{u}))\}_{i=1}^n\big)G , \end{array}$$

as well as

$$M(\mathbf{u}) = (A(\mathbf{u})+\beta B(\mathbf{u}))^{-1}A(\mathbf{u}) .$$

Then $D\mathcal{F}(\mathbf{u}) = M(\mathbf{u})$. We have

$$\mathcal{F}(\mathbf{u}+\boldsymbol{\zeta})-\mathcal{F}(\mathbf{u}) = \int_0^1 D\mathcal{F}(\mathbf{u}+t\boldsymbol{\zeta})\boldsymbol{\zeta}dt = \int_0^1 M(\mathbf{u}+t\boldsymbol{\zeta})\boldsymbol{\zeta}\ dt$$

Using Lemma 3 and H1, for any $\mathbf{v}\in\mathbb{R}^n$, all eigenvalues of $\big(M(\mathbf{v})\big)^TM(\mathbf{v})\big)$ are in $(0,1]$. By the definition of the matrix 2-norm [34] one obtains

$$t\in[0,1] \ \text{ and } \ \boldsymbol{\zeta}\in\mathbb{R}^n \quad\Rightarrow\quad \|M(\mathbf{u}+t\boldsymbol{\zeta})\|_2 \le 1 .$$

Noticing that $M(\cdot)$ is a continuous mapping, the mean value theorem (see e.g. [4], [13]) shows that

$$\|\mathcal{F}(\mathbf{u}+\boldsymbol{\zeta})-\mathcal{F}(\mathbf{u})\|_2 \le \max_{0\le t\le 1}\|M(\mathbf{u}+t\boldsymbol{\zeta})\|_2\ \|\boldsymbol{\zeta}\|_2 \le \|\boldsymbol{\zeta}\|_2 .$$

G. *Proof of Theorem 2*

In order to enhance the dependence of the minimizer function with respect to the regularization parameter $\beta>0$ we shall write $(\mathbf{u},\beta)$ in place of $(\mathbf{u})$.

Since $\mathcal{F}$ is local minimizer function,

$$D_1\mathcal{J}(\mathcal{F}(\mathbf{u},\beta),\mathbf{u};\beta) = 0, \quad \forall\mathbf{u}\in\mathbb{R}^n, \quad \forall\beta>0 , \tag{35}$$

where[9]

$$D_1\mathcal{J}(\mathcal{F}(\mathbf{u},\beta),\mathbf{u};\beta) = D_1\Psi(\mathcal{F}(\mathbf{u},\beta),\mathbf{u}) + \beta D\Phi(\mathcal{F}(\mathbf{u},\beta)) .$$

[9]Note that $D_1\mathcal{J}(\mathcal{F}(\mathbf{u},\beta),\mathbf{u};\beta)$ is the differential of $\mathcal{J}$ with respect to its first argument $\mathcal{F}(\mathbf{u},\beta)$ and that $D\Phi(\mathcal{F}(\mathbf{u},\beta))$ is the differential of Φ with respect to its unique argument $\mathcal{F}(\mathbf{u},\beta)$.

We can thus differentiate with respect to β both sides of (35) which yields

$$0 = D_1^2\Psi(\mathcal{F}(\mathbf{u},\beta),\mathbf{u})\frac{d\mathcal{F}(\mathbf{u},\beta)}{d\beta} + \beta D^2\Phi(\mathcal{F}(\mathbf{u},\beta))\frac{d\mathcal{F}(\mathbf{u},\beta)}{d\beta} + D\Phi(\mathcal{F}(\mathbf{u},\beta)) \ .$$

So the n-length vector $\frac{d\mathcal{F}(\mathbf{u},\beta)}{d\beta}$ reads

$$\frac{d\mathcal{F}(\mathbf{u},\beta)}{d\beta} = -\big(H(\mathbf{u},\beta)\big)^{-1} D\Phi(\mathcal{F}(\mathbf{u},\beta))$$

Using the original expression for Φ in (4), each entry of $D\Phi(\mathcal{F}(\mathbf{u},\beta))$ reads

$$D\Phi(\mathcal{F}(\mathbf{u},\beta))[i] = 2\sum_{j\in\mathcal{N}_i}\phi'(\mathcal{F}_i(\mathbf{u},\beta)-\mathcal{F}_j(\mathbf{u},\beta)) \ .$$

Then

$$\frac{d\mathcal{F}_k(\mathbf{u},\beta)}{d\beta} = -2\sum_{i\in\mathbb{I}_n}\big(H(\mathbf{u},\beta)\big)^{-1}[k,i]\sum_{j\in\mathcal{N}_i}\phi'(\mathcal{F}_i(\mathbf{u},\beta)-\mathcal{F}_j(\mathbf{u},\beta)) \ .$$

References

[1] T. Arici, S. Dikbas, and Y. Altunbasak. A histogram modification framework and its application for image contrast enhancement. *IEEE Transactions on Image Processing*, 18(9):1921–1935, 2009.

[2] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, 2006, Springer-Verlag, Berlin, 2nd edition.

[3] A.N. Avanaki. Exact global histogram specification optimized for structural similarity. *Optical Review*, 16(6):613–621, 2009.

[4] A. Avez. *Calcul Différentiel*. Masson, 1991.

[5] J.-F. Bonnans, J.-C. Gilbert, C. Lemaréchal and C. Sagastizábal. Numerical optimization (theoretical and practical aspects) Springer (Berlin; New York NY; Hong Kong) 2003.

[6] J. Bioucas-Dias, M. Figueiredo, and R. Nowak. Total variation-based image deconvolution: a majorization-minimization approach. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2006*, volume 2, pages II–II, 14–19 May 2006.

[7] M. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications to early vision. *International Journal of Computer Vision*, 19 (1996), pp. 57–91.

[8] N. G. De Bruijn, N. G. Some classes of integer-valued functions. *Indag. Math.*, 17(3), 1955.

[9] P.-J. Cahen and J.-L. Chabert. *Integer-valued Polynomials*. vol. 48 of Mathematical Surveys and Monographs, Amer. Math. Soc., 1997.

[10] R. Chan, M. Nikolova, and Y. Wen. A variational approach for exact histogram specification. *Lecture Notes in Computer Science series*, 6667:preprint, 2011.

[11] S. Chen and A. R. Ramli. Contrast enhancement using recursive mean-separate histogram equalization for scalable brightness preservation. *IEEE Transactions on Consumer Electronics*, 49(4):1301–1309, 2003.

[12] H.D. Cheng, XH Jiang, Y. Sun, and J. Wang. Color image segmentation: advances and prospects. *Pattern recognition*, 34(12):2259–2281, 2001.

[13] P. G. Ciarlet. *Introduction to Numerical Linear Algebra and Optimization*, Cambridge University Press, 1989.

[14] D. Coltuc and P. Bolon. An inverse problem: Histogram equalization. In *Signal Processing IX, Theories and Applications, EUSIPCO'98*, volume 2, pages 861–4, 1998.

[15] D. Coltuc and P. Bolon. Robust watermarking by histogram specification. In *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, volume 2, pages 236–239. IEEE, 1999.

[16] D. Coltuc, P. Bolon, and J.-M. Chassery. Exact histogram specification. *IEEE Transactions on Image Processing*, 15(5):1143–1152, 2006.

[17] I.J. Cox, S. Roy, and S.L. Hingorani. Dynamic histogram warping of image pairs for constant image brightness. In *International Conference on Image Processing*, volume 2, pages 366–369, 1995.

[18] J. Delon. Midway image equalization. *Journal of Mathematical Imaging and Vision*, 21(2):119–134, 2004.

[19] A. L. Dontchev and T. Zollezi. *Well-posed Optimization Problems*. Springer-Verlag, New York, 1993.

[20] L. C. Evans and R. F. Gariepy. *Measure Theory and Fine Properties of Functions*. Studies in Advanced Mathematics, CRC Press, Roca Baton, FL, 1992.

[21] A. Fiacco and G. McCormic. *Nonlinear Programming*. Classics in Applied Mathematics, SIAM, Philadelphia, 1990.

[22] M. A. T. Figueiredo, J. M. Bioucas-Dias and R. D. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Process.*, 16(12):2980–2991, DEC 2007.

[23] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3):367–383, 1992.

[24] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Trans. Image Process.*, 4(7):932–946, JUL 1995.

[25] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Beijlng: Publishing House of Electronics Industry, 2005.

[26] Y. Gousseau and J. M. Morel. Are natural images of bounded variation? *SIAM Journal on Mathematical Analysis*, 33(3):634–648, 2001.

[27] M. Hazewinkel, ed. *Diophantine Equations*. Encyclopaedia of Mathematics, Springer, 2001.

[28] D. Hunter and K. Lange. A tutorial on MM algorithms. *Am. Stat.*, 58(1):30–37, 2004.

[29] A. K. Jain. *Fundamental of Digital Image Processing*. Prentice-Hall, 1989.

[30] H. T. Jongen, P. Jonker, and F. Twilt. *Nonlinear Optimization in Finite Dimensions*. Kluwer Academic Press, 1 ed., 1986.

[31] Y. Kim. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE Transactions on Consumer Electronics*, 43(1):1–8, 1997.

[32] J. Lafontaine. *Introduction aux Variétés Différentielles.* EDP sciences, France, 2 ed., 2010.

[33] D. Krishnan, P. Lin, and X. Tai. An efficient operator splitting method for noise removal in images. *Commun. Comput. Phys.*, 1:847–858, 2006.

[34] C. D. Meyer. *Matrix Analysis and Applied Linear Algebra.* SIAM, 2000.

[35] Y. Nesterov. Smooth minimization of non-smooth functions. *Math. Program., Ser. A*, 103(1):127–152, 2005.

[36] M. Lysaker and X. Tai. Noise removal using smoothed normals and surface fitting. *IEEE Trans. Image Process.*, 13(10):1345–1357, Oct. 2004.

[37] S. Narasimhan and S. Nayar. Contrast restoration of weather degraded images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:713–724, 2003.

[38] K. Ni, X. Bresson, T. Chan, and S. Esedoglu. Local histogram based segmentation using the Wasserstein distance. *International Journal of Computer Vision*, 84(1):97–111, 2009.

[39] M. Nikolova. Local strong homogeneity of a regularized estimator. *SIAM J. on Applied Mathematics*, 61 (2000), pp. 633–658.

[40] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Visio*, 20(1–2):99–120, 2004.

[41] M. Nikolova. Weakly constrained minimization. Application to the estimation of images and signals involving constant regions. *Journal of Mathematical Imaging and Vision*, 21(2):155–175, 2004.

[42] M. Nikolova and M. Ng. Analysis of half-quadratic minimization methods for signal and image recovery. *SIAM J. Sci. Comput.*, 2005.

[43] G. Pólya. Über ganzwertige ganze Funktionen. *Rend. Circ. Mat. Palermo*, 40 (1915), pp. 1-16.

[44] W. K. Pratt. *Digital Image Processing*. 2nd ed., Wiley, New York, 1991.

[45] A.M. Reza. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *The Journal of VLSI Signal Processing*, 38(1):35–44, 2004.

[46] JP Rolland, V. Vo, B. Bloss, and CK Abbey. Fast algorithms for histogram matching: Application to texture synthesis. *Journal of Electronic Imaging*, 9(1):39–45, 2000.

[47] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[48] G. Sapiro and V. Caselles. Histogram modification via partial differential equations. In *Image Processing, 1995. Proceedings., International Conference on*, volume 3, pages 632–635. IEEE, 2002.

[49] C. Vogel and M. Oman. Iterative method for total variation denoising. *SIAM J. Sci. Comput.*, 17:227–238, 1996.

[50] Y. Wan and D. Shi. Joint exact histogram specification and image enhancement through the wavelet transform. *IEEE Transactions on Image Processing*, 16(9):2245–2250, 2007.

[51] C. Wang and Z. Ye. Brightness preserving histogram equalization with maximum entropy: a variational perspective. *IEEE Transactions on Consumer Electronics*, 51(4):1326–1334, 2005.