ORIGINAL PAPER

Variable exponent diffusion for image detexturing

Pierre-Alain Fayolle[1] · Alexander G. Belyaev[2] ●

Abstract

We consider a variational approach to the problem of structure + texture decomposition (also known as cartoon + texture decomposition). As usual for many variational problems in image analysis and processing, the energy we minimize consists of two terms: a data-fitting term and a regularization term. The main feature of our approach consists of choosing parameters in the regularization term adaptively. Namely, the regularization term is given by a weighted $p(\cdot)$-Dirichlet-based energy $\int a(\mathbf{x})|\nabla u|^{p(\mathbf{x})}$, where the weight and exponent functions are determined from an analysis of the spectral content of the image curvature. Our numerical experiments, both qualitative and quantitative, suggest that the proposed approach delivers better results than state-of-the-art methods for extracting the structure from textured and mosaic images, as well as competitive results on image enhancement problems.

Keywords Image decomposition · Structure extraction · Total variation · Variable exponent · Variational formulation

1 Introduction

Given an image, the structure + texture (cartoon + texture) image decomposition refers to the problem of a decomposition of the image intensity function into the sum of two components: the structure component representing homogeneous image regions separated by salient edges and the texture component containing repeating patterns and noise.

Structure + texture image decomposition remains a topic of intensive research due to various applications including low-light image enhancement [1], defect detection [2], image compression [3], image super-resolution [4], and image composition and content-aware image resizing [5]. In applied mathematics, the problem of structure + texture image decomposition led to developing a number of novel variational approaches and numerical algorithms [6, Chapter 1], [7, Section 5.2], [8, Section 15.2], [9, Chapter 5].

In this paper, we attack the structure + texture decomposition problem using a variational approach based on a generalization of the total variation energy. Total variation (TV) regularization was introduced to image processing thirty years ago [10], and since then, it has remained one of the most popular variational regularization technique, and was successively used in a vast majority of image processing tasks. We generalize the TV approach by considering a weighted $p(\cdot)$-Dirichlet-based energy, where the weight and exponent are spatially varying.

1.1 Related work

Several approaches to the problem of structure + texture extraction are based on variational methods and their rigorous mathematical analysis was initiated in [6, Chapter 1]. The image total variation, introduced in [10], was used for texture suppression purposes in [11], [7, Section 5.2], and [9, Chapter 5]. The reader is also referred to [11] for a survey of TV-based structure + texture image decomposition methods. TV-based approaches lead to several variants involving higher order schemes in [12, 13], adaptive TV energies in [5], non-convex regularizations in [14, 15], and low-rank approximations in [16, 17]. Variational approaches with different norms were considered in [6, 18]. A possible issue with TV-based approaches is that they tend to oversmooth the recovered image and blur the main edges.

The variational deep priors approach [19] is an hybrid variational method, where the regularizer (or prior) is learned

✉ Alexander G. Belyaev
a.belyaev@hw.ac.uk

Pierre-Alain Fayolle
fayolle@u-aizu.ac.jp

[1] Department of Information Systems, University of Aizu, Aizu-Wakamatsu, Japan

[2] Institute of Signals, Sensors and Systems, Heriot-Watt University, Edinburgh, UK

from data. More precisely, a family of denoisers parameterized over the noise variance are trained on images corrupted by Gaussian noise with given variance, and used as the regularizer in the variational formulation. The difficulty with this approach is that the family of denoisers may fail to remove the details at a given frequency, since it is not specifically trained for this task.

Several methods based on morphological component analysis (MCA) [8, 20–25] were proposed for extracting the structure of an image. Filter-based methods include the fast nonlinear filter proposed in [26] and a variant using banks of filters [27]. Another filter-based method is the popular guided filter approach introduced in [28]. It may suffer from halo artifacts, an issue that was addressed by considering a gradient domain guided filter in [29]. Variants include [14, 15, 30, 31] or the very recent work [32]. The problem with edge preserving image smoothing approaches is that they are not initially designed to deal with image textures, rely on local properties and may fail to suppress certain high contrast textures.

Other methods include machine learning-based approaches [33–35], approaches based on patch recurrence [36, 37] or the very recent generalized smoothing framework [38, 39]. Finally, it is also possible to extract the texture first. Textures can be characterized by their power spectrum [40]. A power-spectrum-based two-pass approach was very recently introduced in [41]. The difficulty is that in general there is no rigorous or definitive mathematical description of textures [42], and thus, statistical tests on the power spectrum may not be always sufficient for extracting the texture or high-frequency details components.

1.2 Motivation, contribution, and organization

In our treatment of the structure+texture image decomposition problem, we follow the general variational approach and minimize an energy functional consisting of data-fitting and regularization terms (variants of this approach can be found in [7, Section 5.2] and [9, Chapter 5], for example). However, the choice of the regularization term we use in this study has several important differences with the regularization terms commonly used in the structure+texture image decomposition studies. First of all, the conventional variational decomposition approaches mainly focus on selecting texture-insensitive data-fitting terms and the total variation energy constitutes a very popular choice for the regularization term. In contrast to these conventional approaches, our regularization term is data-dependent and imposes strong smoothing in texture-rich image regions. Namely, we consider in this work weighted $p(\cdot)$-Dirichlet energies

$$\int a(\boldsymbol{x})|\nabla u|^{p(\boldsymbol{x})}$$

with spatially varying weight $a(\boldsymbol{x})$ and exponent $p(\boldsymbol{x})$, where the weight and exponent are defined to preserve salient image edges while suppressing the image texture. We propose two schemes for automatically selecting the exponent $p(\boldsymbol{x})$ and the weight $a(\boldsymbol{x})$, one for the problem of structure extraction, where we enforce $0 < p(\boldsymbol{x}) \le 1$, and one for the problem of detail enhancement, where we enforce $1 \le p(\boldsymbol{x}) \le 2$. For both cases, the weight $a(\boldsymbol{x})$ and exponent $p(\boldsymbol{x})$ are determined from an analysis of the spectral content of the level-set and mean curvatures of the image intensity function. Our choice of the level-set curvature spectral content is motivated by its sensitivity to both the low-contrast and high-contrast textures. On the other hand, the mean curvature is much more numerically stable than the level-set curvature and, therefore, the use of image mean curvature is preferable when dealing with detail enhancement tasks. Similarly, setting $0 < p(\boldsymbol{x}) \le 1$ leads to aggressive image texture smoothing, while choosing $1 \le p(\boldsymbol{x}) \le 2$ yields more gentle smoothing which is appropriate for image sharpening purposes through the standard unsharp-masking procedure.

Unlike the methods from [43] and [44], we use a single pass of an ADMM-based optimization procedure, making the proposed approach more efficient.

Our numerical experiments suggest that the proposed approach delivers better results than state-of-the-art methods for extracting the structure from textured images and mosaic images, as well as competitive results on image enhancement problems.

The rest of this work is organized as follows: We describe our approach in Sect. 2. The results of our numerical experiments are provided in Sect. 3, as well as additional possible applications of the proposed approach. Finally, we conclude in Sect. 4.

2 Proposed approach

2.1 A visual illustration of the approach

We start with summarizing the whole approach. It is illustrated visually in Fig. 1. Given an input image, in Fig. 1a, we want to remove the texture and extract the main structure of the image, as in Fig. 1d. First, we compute the level-set curvature (3) of the image as shown in Fig. 1b, or the surface mean curvature (4), which is then used to define a disparity measure (6). Next, we use this disparity measure to define a weight function (shown in the top row of Fig. 1c), and an exponent function (shown in the bottom row of Fig. 1c). Finally, the image structure, in Fig. 1d, is obtained by minimizing the weighted $p(\cdot)$-Dirichlet energy (1), or (2), defined in terms of these weight and exponent functions.

Fig. 1 A visual illustration of the approach: **a** the input image; **b** the level-set curvature (3); **c** top image: the weight function $a(\cdot)$; Bottom image: the exponent function $p(\cdot)$; **d** the extracted structure

(a) Input image   (b) Curvature   (c) Weight/exponent   (d) Image structure
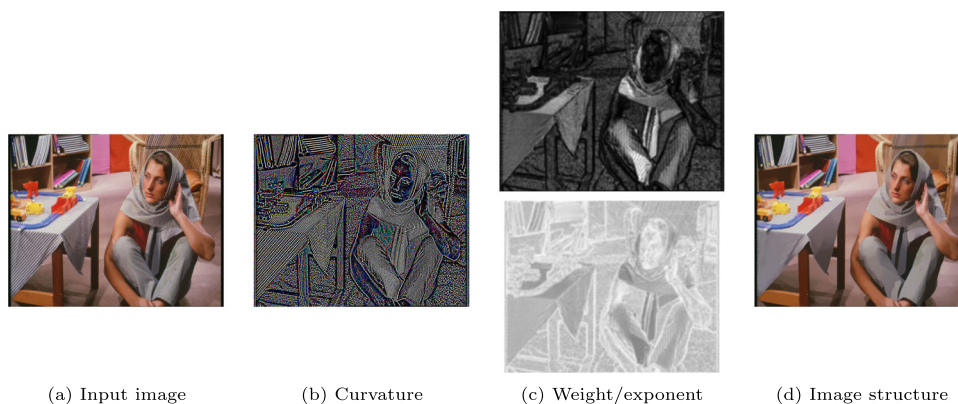
2.2 Weighted p-Dirichlet energy

Given a bounded domain $\Omega \subset \mathbb{R}^2$, and a function f defined on Ω,[1] we consider the following weighted $p(\cdot)$-Dirichlet energy

$$\int_\Omega \left(a(\boldsymbol{x})|\nabla u|^{p(\boldsymbol{x})} \right) + \frac{\lambda}{2} \int_\Omega (H[u-f])^2 \longrightarrow \min, \qquad (1)$$

with variable exponent $p(\boldsymbol{x})$ and weight function $a(\boldsymbol{x})$. H is a smoothing operator, and we use the convolution with a Gaussian

$$H[u-f] = G_\sigma * (u-f), \quad G_\sigma(\boldsymbol{x}) = G(\boldsymbol{x}/\sigma)/\sigma^2,$$

where, unless explicitly mentioned, $\sigma = 1$. An alternative to (1) consists in using the L_1 norm for the fitting term, leading to

$$\int_\Omega \left(a(\boldsymbol{x})|\nabla u|^{p(\boldsymbol{x})} \right) + \lambda \int_\Omega |H[u-f]| \to \min. \qquad (2)$$

The solution u to the problem (1), respectively (2), extracts the main structure of the original image f. Natural and structure images are usually sparse in the gradient domain, thus the consideration of the L_p-based gradient regularization term $\int_\Omega |\nabla u|^{p(\boldsymbol{x})}$ with an exponent $0 < p(\boldsymbol{x}) \le 1$, which promotes sparsity.

In (1), respectively (2), the first term enforces sparseness of the resulting image, while the second term corresponds to a fitting term that enforces similarities with the input image. The weight λ controls the trade-off between these two terms. In practice, for the structure extraction problem, we would like to assign more weight to the first term in the textured regions, and a low weight in the non-textured regions, thus the introduction of a spatially varying weight $a(\boldsymbol{x})$ for the first term. Similarly, using a spatially varying exponent $p(\boldsymbol{x})$ allows us to control the gradient sparsity of the structure

image $u(\boldsymbol{x})$, by using low values for $p(\boldsymbol{x})$ in textured regions. We discuss further the expressions used for $a(\boldsymbol{x})$ and $p(\boldsymbol{x})$ in Sect. 2.4.

The $p(\cdot)$-Dirichlet energy term (1) with variable exponent $p(\boldsymbol{x})$ was previously considered in [43–47]. The case $p(\boldsymbol{x}) \ge 1$ was considered in [45–47]. It leads to a convex energy for which more results and tools are available. Similarly to [43, 44], we consider for structure extraction the case with $0 < p(\boldsymbol{x}) \le 1$, for which fewer results exist. The case $p \equiv \text{const}$, $p \in [0, \infty)$ was studied in [48, 49], where the weighted graph p-Laplacian is applied to problems in image and mesh processing. Indeed, for $p \in [2, \infty)$

$$\Delta_p u = \text{div}(|\nabla u|^{p-2}\nabla u) = 0$$

is the Euler–Lagrange equation to $\int_\Omega |\nabla u|^p$.

One problem with the case $p < 1$ is that the energy (1) is non-convex, and thus, the uniqueness of the minimizer is not guaranteed. In practice, we minimize (1) with the alternating direction method of multipliers (ADMM) [50], see also [51], and found experimentally that the obtained minimizer delivered very good results. The energy (2) is non-convex as well, when $p < 1$, and similarly dealt with by ADMM.

2.3 Minimization of the weighted $p(\cdot)$-Dirichlet energy

We use a variant of ADMM to solve numerically (1). The problem (1) can be rewritten as

$$\int_\Omega a(\boldsymbol{x})|\boldsymbol{\xi}|^{p(\boldsymbol{x})} + \frac{\lambda}{2} \int_\Omega (H[u-f])^2 \to \min$$

with the constraint

$$\boldsymbol{\xi} = \nabla u.$$

---

[1] Ω is a rectangle for images.

The minimization is performed by the repetition of the following steps

$$\boldsymbol{\xi}_{k+1} \leftarrow \max \left\{ |s_k| - \frac{a(\boldsymbol{x})}{r |\nabla u_k|^{1-p(\boldsymbol{x})}}, 0 \right\} \frac{s_k}{|s_k|},$$

$$s_k = \nabla u_k - \frac{\boldsymbol{\mu}_k}{r},$$

$$u_{k+1} \leftarrow \text{Solution of } - r(\Delta u - \text{div}(\boldsymbol{\xi}_{k+1}))$$
$$+ \text{div}(\boldsymbol{\mu}_k) + \lambda H^* H[u - f] = 0,$$

$$\boldsymbol{\mu}_{k+1} \leftarrow \boldsymbol{\mu}_k + \rho \left(\boldsymbol{\xi}_{k+1} - \nabla u_{k+1} \right).$$

See "Appendix A" for a derivation of these steps. A similar approach is used for solving numerically (2). See "Appendix B" for details.

One of the remaining issues to be resolved is the definition of the functions for the exponent $p(\boldsymbol{x})$ and the weight function $a(\boldsymbol{x})$. We propose to learn them from information gathered from the input image. In particular, we use the curvature information.

2.4 Computation of $p(x)$ and $a(x)$

For the image structure extraction problem, it seems reasonable to force the weight $a(\boldsymbol{x})$ in (1), or (2), to be large for the small-scale image details and texture regions of the image that we want to suppress, and low, otherwise. It corresponds to assigning more weight in the textured regions to the first term of (1), whose task is to enforce gradient sparsity of $u(\boldsymbol{x})$. Similarly, the exponent $p(\boldsymbol{x})$ controls the gradient sparsity of the structure image $u(\boldsymbol{x})$, thus it makes sense to force low values in textured regions. Perfect expressions for $a(\boldsymbol{x})$ and $p(\boldsymbol{x})$ are not known in general, and we need to estimate them from the input image using, for example, the curvature information.

We can assume that small-scale image details and texture contribute to the high-frequency part of the level-set curvature

$$k(\boldsymbol{x}) = \text{div} \left(\frac{\nabla f(\boldsymbol{x})}{|\nabla f(\boldsymbol{x})|} \right). \tag{3}$$

One possible issue with the level-set curvature $k(\cdot)$ is that it is difficult to estimate it robustly in some cases. Indeed, for natural images with less texture (consider for example a perfectly blue sky), $|\nabla f(\boldsymbol{x})|$ is close to 0, making (3) ill-defined. An illustration of this fact is shown in Fig. 2. An alternative is to view the image as the height field $z = f(\boldsymbol{x})$

and to consider instead the surface mean curvature[2]

$$K_m = \frac{1}{2} \text{div} \left(\frac{\nabla f}{\sqrt{1 + |\nabla f|^2}} \right). \tag{4}$$

For images with lots of structure such as, for example, mosaics, using the level-set curvature (3) is preferable. However, for images containing less texture, it is better to use the mean curvature (4) to avoid visual artifacts.

Let $\tilde{k}(\boldsymbol{x})$ be obtained from $k(\boldsymbol{x})$ (or similarly from $K_m(\boldsymbol{x})$ when the mean curvature is used instead) by suppressing high frequencies of $k(\boldsymbol{x})$ (respectively, $K_m(\boldsymbol{x})$). In practice, we use the Butterworth low-pass filter

$$\left[1 + \left(\frac{|\omega|}{b} \right)^m \right]^{-1} \tag{5}$$

with cutoff frequency b. We estimate the difference between k and $\tilde{k}$ by $|k - \tilde{k}|$, smooth it by convolution with the Gaussian kernel G_σ ($\sigma = 3$ was used in our experiments) and normalize it to obtain the disparity

$$d(\boldsymbol{x}) = \frac{G_\sigma * |k - \tilde{k}|}{\max \left(G_\sigma * |k - \tilde{k}| \right)}, \tag{6}$$

where, by construction, $0 \le d(\boldsymbol{x}) \le 1$.

Under the assumption that the small-scale details and texture contribute to the high-frequency content of the curvature, the disparity function $d(\boldsymbol{x})$ is a normalized function, which is close to 0 in non-textured regions (where the curvature k and the filtered curvature $\tilde{k}$ agree), and close to 1 in textured regions (where the curvature k is high, and the filtered curvature $\tilde{k}$ low). This disparity function can be used for providing expressions for the spatially varying weight $a(\boldsymbol{x})$ and exponent $p(\boldsymbol{x})$.

Structure extraction

For the problem of structure extraction, we define the weight $a(\boldsymbol{x})$ and the variable exponent $p(\boldsymbol{x})$ from the disparity $d(\boldsymbol{x})$ as follows

$$a(\boldsymbol{x}) = d(\boldsymbol{x})^2 \quad \text{and} \quad p(\boldsymbol{x}) = 1 - \frac{d(\boldsymbol{x})}{2}. \tag{7}$$

As stated earlier, our goal is to set $a(\boldsymbol{x})$ to be large in textured regions (i.e., to give more weight to the regularization term and enforce gradient sparsity) and low in non-textured regions. Removing the high frequency curvature content affects corners and some other salient image features. So

---

[2] We use K_m for the mean curvature instead of the usual symbol H, which is already used to denote a smoothing operator.

Fig. 2 An example of natural image is shown with the computed level-set curvature (3) and the mean curvature (4), scaled by a factor 50. Notice the noise in the region where the level-set curvature is close to 0 (the blue sky)

Input image   Level-set curvature (3)   Mean curvature (4)
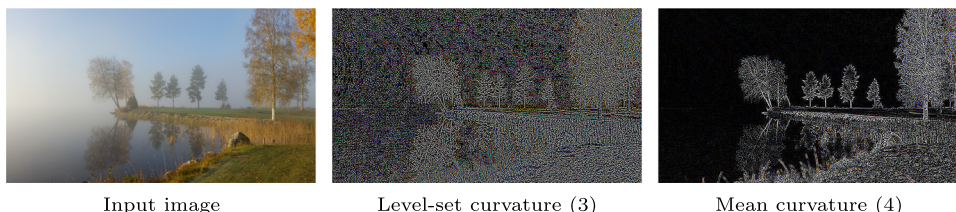
if $d(\boldsymbol{x})$ is small, it is better to reduce the smoothing effect of the $p(\cdot)$-Dirichlet term. A simple choice for such a reduction is $a(\boldsymbol{x}) = d(\boldsymbol{x})^2$. For the expression of the exponent $p(\boldsymbol{x})$, a constant equal to 1 would correspond to the TV model [10], which preserves image edges. Using $p(\boldsymbol{x}) < 1$ further enforces (gradient) sparsity. Lower values of $p(\boldsymbol{x})$ should be obtained in textured regions, where the normalized disparity $d(\boldsymbol{x})$ is close to 1.

Image sharpening

For the problem of image sharpening, we use the following variant

$$a(\boldsymbol{x}) = d(\boldsymbol{x})^2 \quad \text{and} \quad p(\boldsymbol{x}) = 2 - d(\boldsymbol{x}). \tag{8}$$

In the case of image detail sharpening, we use the same expression for $a(\boldsymbol{x})$ with a similar justification. When we deal with the image sharpening problem, we need an accurate smoothing method preserving salient edges. The choice $p(\boldsymbol{x}) < 1$ is too aggressive for this purpose, so we choose $p(\boldsymbol{x})$ such that $1 < p(\boldsymbol{x}) < 2$, justifying our choice for (8).

3 Numerical experiments and applications

We compare the approach described in this work to recent approaches for the problem of structure + texture extraction. A direct application is the extraction of the structure image from a mosaic image. We also consider another possible application of image structure extraction: sharpening details in images. For each of these problems, we compare the proposed approach to recent and state-of-the-art methods.
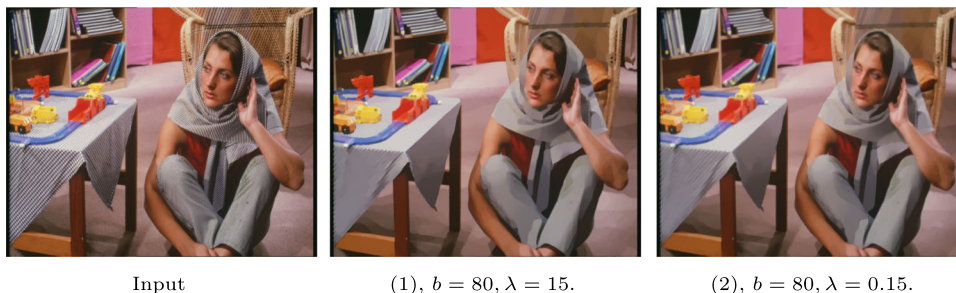
3.1 Structure + texture decomposition

When considering the problem of structure + texture decomposition, the goal is to extract the structure u from an input image f, the texture component is then given by $f - u$. We compute u as a minimizer of the $p(\cdot)-$Dirichlet energy with an L_2 (1) or L_1 fitting term (2) using the numerical method described in Sect. 2. We use the left image in Fig. 3 as a first test image. The middle and right images in Fig. 3 illustrate the results obtained by our approach when using the L_2 fitting term or the L_1 fitting term, respectively.

A visual comparison between our approach, [19, 32, 39, 41] is provided in Fig. 4, where zooms on two selected regions are also shown. These selected regions are further magnified in Fig. 5. We can see that the methods from [32] and [19] tend to over-smooth the structure, while the method from [39] fails to remove the texture from the napkin properly, see the zoomed regions and Fig. 5, as well. Visually, the best results are obtained by our approach and the method from [41]. The zoomed regions in Figs. 4 and 5 demonstrate that both approaches retain the main structure of the face, while removing the texture from the napkin.

To better appreciate the results obtained with the method from [41] and our approach, we show in Fig. 6 the one dimensional signals corresponding to a cross section of Fig. 6. The cross section is done along the line in blue color in Fig. 6, left image. The top-most graphs correspond to the initial signal, while the graphs shown in the middle correspond to a slice of the structure obtained from our approach (the middle image) and the method from [41] (the right image). Ultimately, we want to preserve the low frequency components of the signal, without over-smoothing the sharp features or shrinking the signal too much, and while removing the high frequency components. The difference between the two signals is shown

Fig. 3 Structure extraction from a test image (left). The extracted image structure obtained by our approach is shown in the middle (L_2 fitting term is used (A1)) and right images (L_1 fitting term is used (B3))

Input   (1), $b = 80, \lambda = 15$.   (2), $b = 80, \lambda = 0.15$.

Fig. 4 Structure extracted from Fig. 3, left image, for different approaches

Original image   (1), $b = 80, \lambda = 15$ [39]

[32]   [41]   [19]
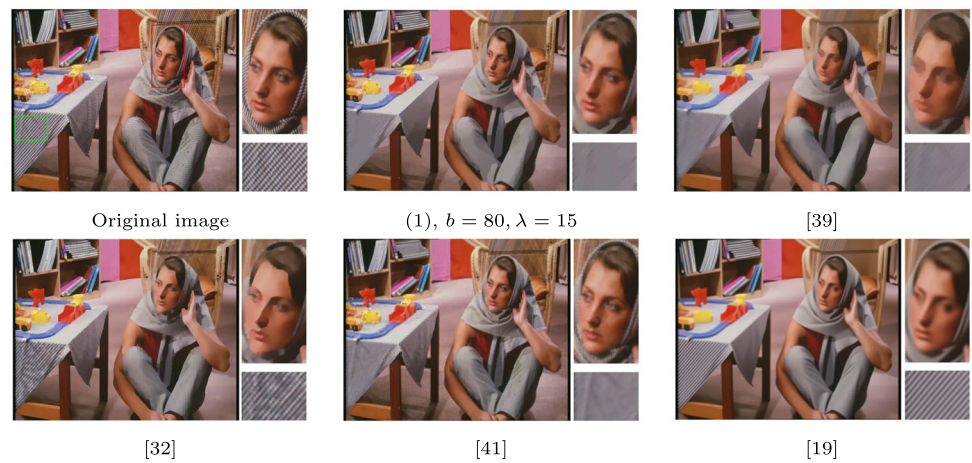
Fig. 5 The zoomed regions from Fig. 4

(1), $b = 80, \lambda = 15$ [39]   [32]   [41]   [19]

Initial image   (1), $b = 80, \lambda = 15$ [41]
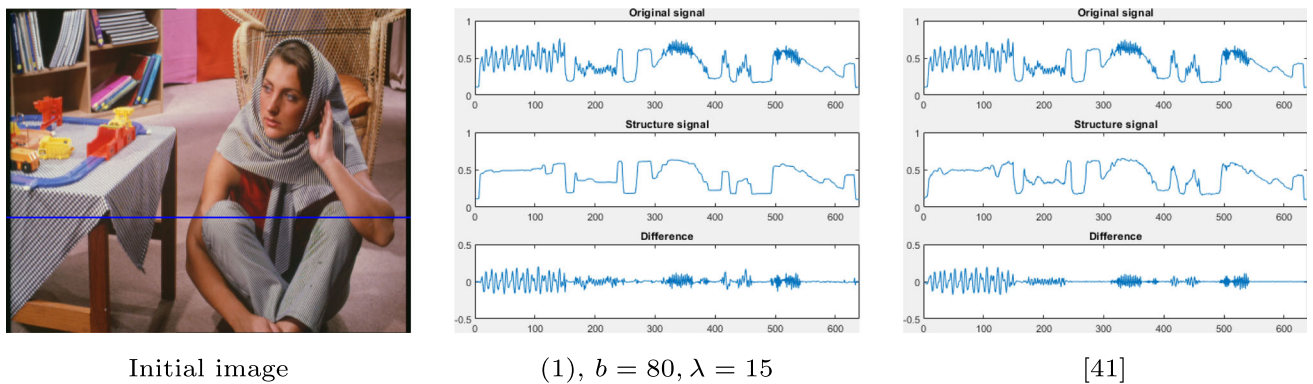
Fig. 6 Cross section along the blue line (left image) for some of the structure images shown in Fig. 4

in the bottom-most graphs. It corresponds to the texture information of the image (the high-frequency components).

L_2 against L_1 fitting term

One could expect the L_1 fitting term to lead to a better fit than the L_2 fitting term in the regions of low disparity and thus a better preservation of the main edges of the image structure. In practice, however, we did not find a large difference between the results obtained when a L_2 fitting term (1) or a L_1 fitting term (2) are used. Figure 3 illustrates the results obtained in each case. A zoom on the selected regions shown in Fig. 4 is shown in Fig. 7.

| (1) | (2) |

Fig. 7 A zoom on selected regions shown in Fig. 4 when the L_2 (1), respectively, L_1 (2) are used

Parameters sensitivity

The proposed approach depends on two main parameters: b, the cutoff frequency of the Butterworth filter (5), and λ, the weight controlling the trade-off between the fitting term and the regularization term in (1), respectively (2). Intuitively, lower values of b suppress more terms in the spectrum of the filtered curvature $\tilde{k}$; thus, it forces the suppression of the image edges. Consequently, b should be set low enough to suppress texture, but not too low; otherwise, it tends to smooth the main image edges as well. The parameter λ controls the weight of the fitting term against the regularization term. Higher values for λ force the image structure u to stay closer to the input image f. The sensitivity of the proposed approach to these two parameters is illustrated in Fig. 8 when the L_2 fitting term is used, and in Fig. 9 when the L_1 fitting term is used.

Ablation study

In Fig. 10, we study the impact of the spatially varying weight function $a(x)$ and the spatially exponent function $p(x)$, by individually setting them to a constant value. In (a), the proposed approach (1) with the parameter values $b = 80, \lambda = 15$ is used. In (b), the weight function is set to a constant $a(x) = 1$, while the exponent function $p(x)$

is kept varying. The result appears to be oversmoothed and the main features of the structure image are not preserved. In (c) and (d), the weight function $a(x)$ is spatially varying, while the exponent function $p(x)$ is kept constant to the values 1/2 and 1, respectively. When $p(x) = 1/2$, the result is too smooth overall, though it appears to be sharper than (b). The case $p(x) = 1$ allows to preserve sharper edges for the face (top row), but cannot filter the texture on the napkin (bottom row). Overall, (a) delivers the best result.

3.2 Extracting a structure image from a mosaic image

A direct application of structure extraction methods is used for dealing with the problem of extracting the main structure image from a mosaic image. Several examples of mosaic images are shown in Figs. 11, 12 and 13. These mosaic images are processed by our approach, and the methods from [19, 32, 39, 41]. While the method from [41] performed very well for extracting the image structure from a textured image (see Figs. 4, 5, 6), it is surprisingly having difficulties with mosaic images. In most cases, it fails at removing the tile patterns. The method from [19] performs very well on several of these mosaic images, but is also significantly slower, and can over-smooth parts of the images sometimes. Our approach performs similarly well: On some images, it delivers slightly worse results than [19], while on others, it does a better job at preserving some of the main image details.

3.3 Quantitative analysis

In general, there are no widely used metrics or quantitative methods for comparing the output of structure+texture decomposition methods and we have to rely essentially on visual and qualitative comparisons, as done in the previous sections.

One possible approach for a quantitative analysis of image structure extraction methods could be as follows: generate a synthetic structure image s (for example, a piecewise constant function), load a texture image t from a data-set, such as, for example, [52] and form the image $s + t$ (with appropriate weighting, if necessary). Then, one can compute a structure image s_1 with a given structure extraction method and compare it with the known structure s using the peak signal-to-noise ratio (PSNR) or the structural similarity index measure (SSIM). We will use the simple example from Fig. 14 to show that this approach cannot work in general.

Figure 14 shows a synthetic textured image (st) and two possible decompositions into structure and texture: $st = s_1 + t_1 = s_2 + t_2$. The structure (respectively, texture) images differ only by a constant shift (i.e., $s_1 = s_2 + c$, where c is a constant). In this example, the synthetic image

Fig. 8 Influence of the two main parameters b and λ when minimizing (1)
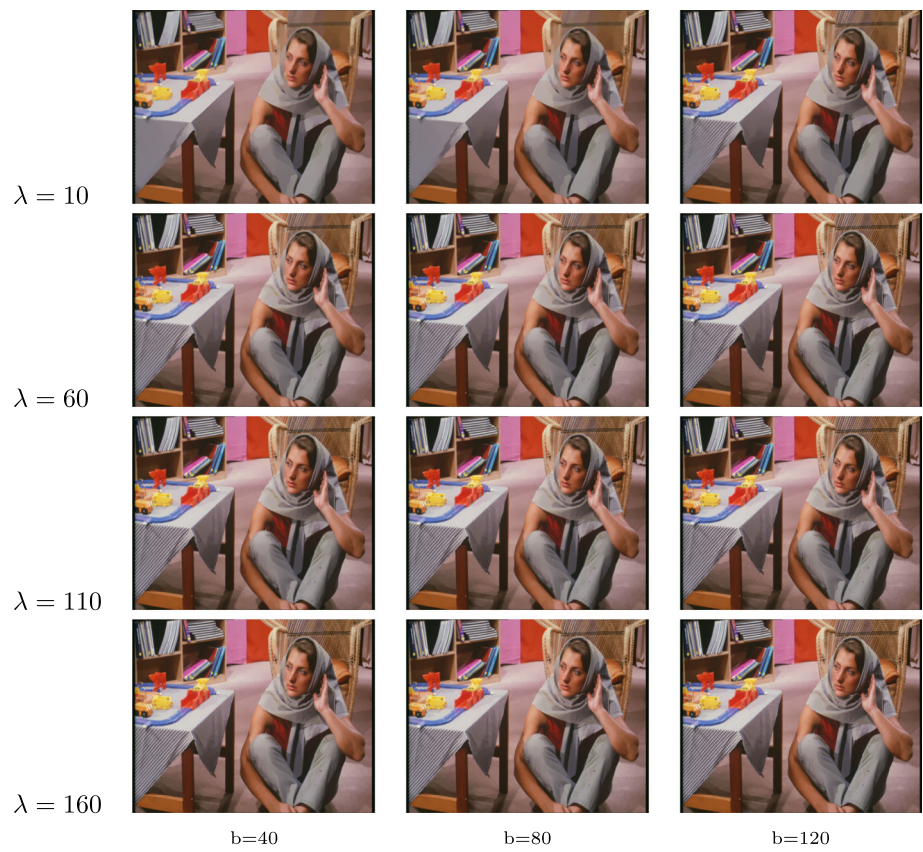

$\lambda = 10$

$\lambda = 60$

$\lambda = 110$

$\lambda = 160$

b=40   b=80   b=120

Fig. 9 Influence of the two main parameters b and λ when minimizing (2)

$\lambda = 0.05$

$\lambda = 0.1$

$\lambda = 0.5$

$\lambda = 1.0$

b=40   b=80   b=120
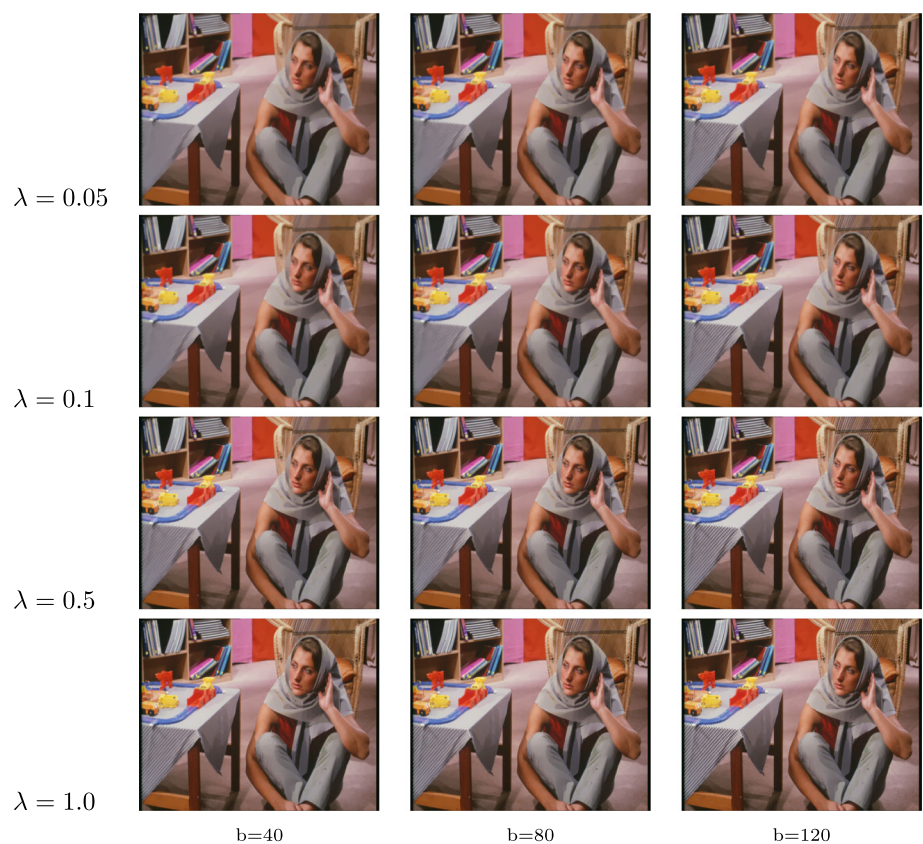
Fig. 10 Influence of the spatially varying weight $a(\boldsymbol{x})$ and exponent $p(\boldsymbol{x})$ functions on the result. **a** (1) is minimized with the parameters $b = 80$ and $\lambda = 15$. In **b**, **c** and **d** either the weight function $a(\boldsymbol{x})$ or the exponent function $p(\boldsymbol{x})$ is set to a constant

(a) (1), $b = 80, \lambda = 15$ (b) $a(\boldsymbol{x}) = 1, p(\boldsymbol{x})$ (c) $a(\boldsymbol{x}), p(\boldsymbol{x}) = 1/2$ (d) $a(\boldsymbol{x}), p(\boldsymbol{x}) = 1$

Fig. 11 Extracting the structure from mosaic images via different approaches

Input mosaic   (1), $b = 30, \lambda = 10$ [39]

[32]   [41]   [19]

Input mosaic   (1), $b = 30, \lambda = 10$ [39]

[32]   [41]   [19]

Fig. 12 Extracting the structure from mosaic images via different approaches

| Input mosaic | (1), $b = 30, \lambda = 6$ | [39] |
| [32] | [41] | [19] |

| Input mosaic | (1), $b = 30, \lambda = 10$ | [39] |
| [32] | [41] | [19] |
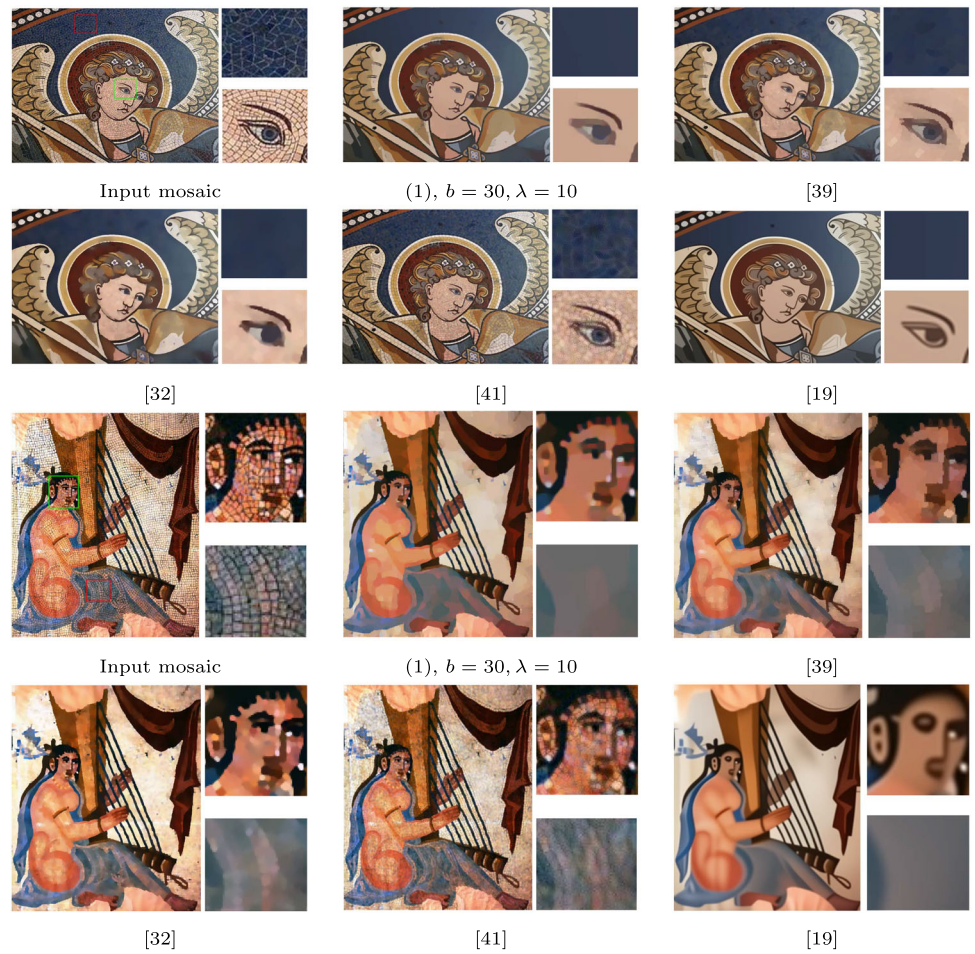
Fig. 13 Image structure extraction from a mosaic image via different approaches

| Input mosaic | (1), $b = 50, \lambda = 15$ | [39] |
| [32] | [41] | [19] |
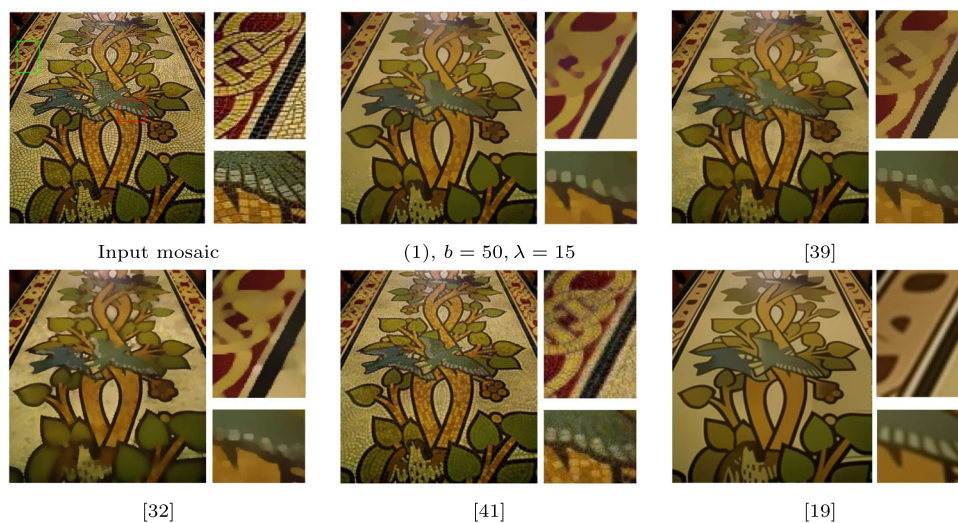
is obtained from the sum of the first structure and texture (s_1 and t_1). The corresponding scores are: $\text{PSNR}(s_1, s_2) = 16.5$, $\text{SSIM}(s_1, s_2) = 0.71$, and are relatively low. Additionally, we have: $\text{PSNR}(st, s_1) = 16.31$, $\text{SSIM}(st, s_1) = 0.68$, and $\text{PSNR}(st, s_2) = 41.6$, $\text{SSIM}(st, s_2) = 0.96$. This synthetic example illustrates that the problem is ill-posed and that a quantitative analysis on a synthetic data-set is difficult to perform in general.

In spite of these difficulties, a recent work [53] proposed a method for trying to quantify the performance of structure extraction methods. The approach is interactive and relies on the user to select points on an extracted structure image. These points are classified in two separate categories: The structure category and the texture category. Scores are computed with different methods for the neighborhoods of points in each category and combined by cross-entropy. Intuitively, neighborhoods of points in the structure category should be
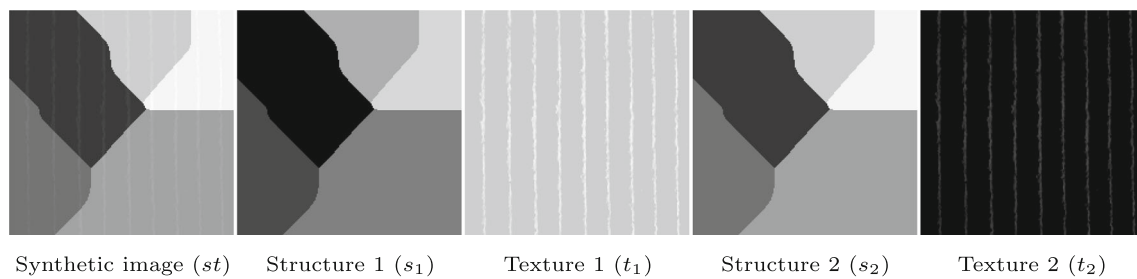
| Synthetic image (st) | Structure 1 (s_1) | Texture 1 (t_1) | Structure 2 (s_2) | Texture 2 (t_2) |

Fig. 14 An example illustrating the difficulty of performing a quantitative analysis. A synthetic image is obtained from the sum of the first structure and texture. The second structure and texture (obtained by a constant shift) yield the same synthetic image. Note that the textures are scaled by 5 for visibility purposes

similar to the input image, and neighborhoods of points in the texture category should have the texture smoothed (they should not preserve edges). This method does not seem to be widely used so far, outside of the work [32], possibly because it relies on a user to select key points. Nevertheless, we provide in Table 1 the scores obtained with this approach on the images used in Figs. 4, 11, 12, and 13.

3.4 Running time

The proposed approach was implemented in MATLAB.[3] The running times for the structure extraction from the textured image (Fig. 4), and from the mosaic images in Figs. 11, 12 and 13 are provided in Table 2. These running times were obtained on an Intel CORE i3 at 3.3GHz using a single thread and are averaged over multiple runs. All the other methods used in the comparison [32, 39, 41] are similarly implemented in MATLAB and run in the same conditions. The computational timings for [19] are omitted from this table, because they are not directly comparable to the other methods. The approach relies on a denoiser implemented by a deep neural network and is not competitive when run on a CPU. One can observe that our schemes show competitive computational times, while providing high quality results.

The most time-consuming part (approximately 25% of the total time) in our approach is the solution at each iteration of a simple linear PDE in the second step of the minimization procedure described in Sect. 2.3. This PDE is solved in frequency space using the fast Fourier transform.

All these approaches, with the exception of [41], are iterative, and run for a sufficient number of iterations to compute the image structure. For energy-based methods, such as ours, the energy can also be monitored and used as a stopping criterion.

The time-consuming part in the method described in [32] is the computation at each iteration of the edge aware filters and their application, while the method [39] needs to solve

a linear system at each iteration by the pre-conditioned conjugate gradient method. The method described in [41] spent most of its time is in computing statistics on image patches.

3.5 Image details sharpening and enhancement

Another domain of application is image sharpening. Given an input image f, we extract the structure u using the approach described in Sect. 2 and then compute the texture $t = f - u$. We use a simple method to sharpen the details: We simply multiply the texture by a constant. The sharpened image (f_S) is finally obtained by $f_S = u + \alpha(f - u)$. In the results shown in Fig. 15, we set $\alpha = 5$. For comparison, we show also the results obtained by combining the approach previously described for image sharpening with different state-of-the-art approaches [32, 39, 41] for the structure extraction. A zoom on the regions highlighted in red and green color in the top-left image of Fig. 15 is shown in Fig. 16. Both methods from [39] and [32] generate artifacts, which are highlighted by the white color boxes in Fig. 16. The method from [41] and our approach do not generate such artifacts while sharpening the image details.

Enhancing images with haze or mist is another example of possible application. An example of an image and its processed versions is shown in Fig. 17, with zooms on selected regions shown in Fig. 18. The process is relatively similar to image sharpening: given an observed image, its structure is extracted and the recovered details are amplified. When dealing with natural images, we saw in Sect. 2.4 that the surface mean curvature (4) was more stable than the level-set curvature (3). This is illustrated in the middle and right images in Fig. 17, where the level-set curvature, respectively, the surface mean curvature, is used for the computation of the disparity term (6). We can observe in the middle image visual artifacts coming from the use of the level-set curvature, and amplified by the image enhancement process. The image on the right does not have these artifacts. The zoomed regions in Fig. 18 indicate that both methods are able to enhance image details despite the haze.

---

[3] The code is available at https://github.com/fayolle/tv_ap_onepass_demo.

Table 1 T3SI scores [53] (Higher value, shown in bold font, is better)

	ADMM for (1)	[32]	[39]	[41]	[19]
Figure 4	**2.62**	2.44	2.53	2.45	2.48
Figure 11—top row	**2.67**	2.65	2.59	2.38	2.62
Figure 11—bottom row	**2.77**	2.61	2.71	2.31	2.70
Figure 12—top row	**2.48**	2.47	2.20	2.25	2.28
Figure 12—bottom row	**2.70**	2.58	2.48	2.18	2.53
Figure 13	**2.70**	**2.70**	2.66	2.31	**2.70**

Table 2 Running times (in sec.) for the structure extraction from textured and mosaic images

	ADMM for (1)	ADMM for (2)	[32]	[39]	[41]
Figure 4	6.86	7.88	23.65	14.62	36.05
Figure 11—top image	10.87	13.13	34.17	22.14	52.30
Figure 11—bottom image	6.00	7.26	18.03	10.96	27.42
Figure 12—top image	12.42	14.72	35.37	21.32	54.32
Figure 12—bottom image	5.91	6.77	17.27	9.80	25.80
Figure 13	6.33	7.35	19.13	12.23	28.99

Input	(1) with (3), $b = 80$, $\lambda = 5$	(1) with (4), $b = 80$, $\lambda = 5$
[39]	[32]	[41]

Fig. 15 Applications to image sharpening. An input image is sharpened by proper re-scaling of the texture information. Different methods for computing the image structure (and thus the texture) are compared

4 Discussion and conclusion

We have presented in this work an approach for extracting the structure and texture of an image. The approach is based on minimizing a weighted version of a $p(\cdot)$-Dirichlet energy with spatially variable weight and exponent functions. Our choices of the weight and exponent functions are adapted for the tasks considered in the paper: structure extraction and image details sharpening. In both cases, we construct the functions based on an analysis of the spectral contents of the level-set or mean curvature of the image intensity.

For the structure extraction problem, we demonstrated experimentally the advantages of the proposed approach over recent state-of-the-art structure + texture decomposition methods on textured images and mosaic images. In particular, we have demonstrated that our approach does a good job at preserving the main edges and appearance of the input image, while removing high frequency details.

Using a recent method [53], we have tried to quantify the result of these different methods on the structure extraction task. However, we have also shown with a simple synthetic example that the problem of structure extraction is ill-posed,
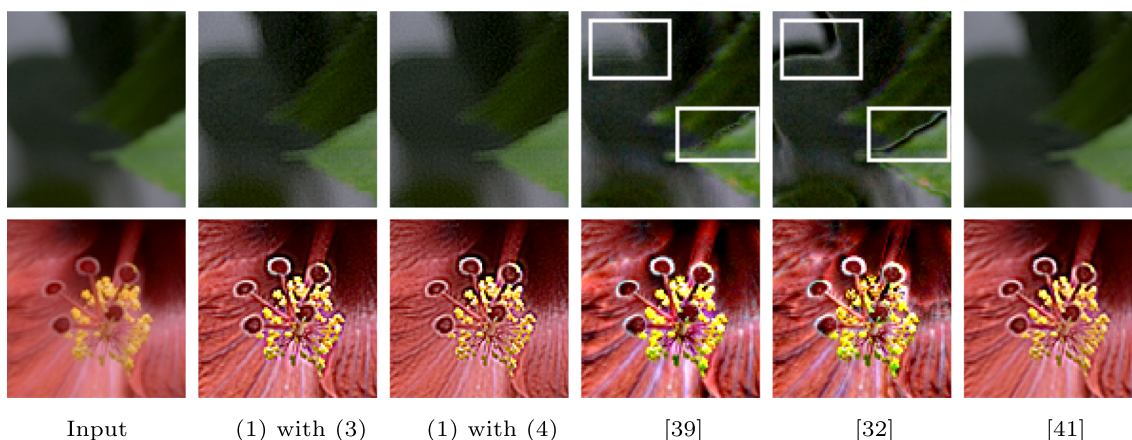
| Input | (1) with (3) | (1) with (4) | [39] | [32] | [41] |

Fig. 16 Zooms on selected regions of the sharpened images in Fig. 15

Fig. 17 An example of enhancing an image with mist. Left: The original image; middle: Enhanced image with the level-set curvature (3) used for the structure extraction; right: enhanced image with the surface mean curvature (4) used for the structure extraction. In both cases, $b = 50$, $\lambda = 1$ are used

Fig. 18 Zooms on selected regions of the dehazed images in Fig. 17

| Input | (1) with (3) | (1) with (4) |

and that a quantitative analysis is difficult to do in general, and that the results are difficult to interpret.

Finally, we have demonstrated a few possible applications, such as sharpening image details, and enhancing an image with mist or haze. The former illustrated the advantages of our approach against other recent methods, while the latter was used to demonstrate the advantage of using the surface mean curvature for determining the weight and exponent functions of the $p(\cdot)$-Dirichlet energy.

Declarations

Conflict of interest The authors declare no competing interests.

Code availability The code is available at https://github.com/fayolle/tv_ap_onepass_demo.

Appendix A Minimization by ADMM

We use a variant of ADMM to solve numerically (1) and (2). The problem (1) can be rewritten as

$$\int_\Omega a(\boldsymbol{x})|\boldsymbol{\xi}|^{p(\boldsymbol{x})} + \frac{\lambda}{2}\int_\Omega (H[u-f])^2 \to \min$$

with the constraint

$$\boldsymbol{\xi} = \nabla u.$$

The corresponding augmented Lagrangian is

$$L_{a,p}(u,\boldsymbol{\xi},\boldsymbol{\mu}) = \int_\Omega \Big\{ a(\boldsymbol{x})|\boldsymbol{\xi}|^{p(\boldsymbol{x})} + \frac{\lambda}{2}H[u-f]^2 \\ + \boldsymbol{\mu}\cdot(\boldsymbol{\xi}-\nabla u) + \frac{\rho}{2}|\boldsymbol{\xi}-\nabla u|^2 \Big\}, \qquad \text{(A1)}$$

where $\boldsymbol{\mu}(\boldsymbol{x})$ is the vector of Lagrange multipliers and $\rho > 0$ is a constant. The Lagrangian (A1) is minimized by block descent and updating in turns $\boldsymbol{\xi}$, u and $\boldsymbol{\mu}$

$$\boldsymbol{\xi}_{k+1} = \arg\min L_{a,p}(u_k, \boldsymbol{\xi}, \boldsymbol{\mu}_k),$$
$$u_{k+1} = \arg\min L_{a,p}(u, \boldsymbol{\xi}_{k+1}, \boldsymbol{\mu}_k),$$
$$\boldsymbol{\mu}_{k+1} = \arg\min L_{a,p}(u_{k+1}, \boldsymbol{\xi}_{k+1}, \boldsymbol{\mu}).$$

There is no closed-form solution for the first step. Instead, we resort to an approximation for the pth power of the L_p norm of $\boldsymbol{\xi}$

$$\boldsymbol{\xi}_{k+1} = \max\left\{|\boldsymbol{s}_k| - \frac{a(\boldsymbol{x})}{r|\nabla u_k|^{1-p(\boldsymbol{x})}}, 0\right\}\frac{\boldsymbol{s}_k}{|\boldsymbol{s}_k|},$$
$$\boldsymbol{s}_k = \nabla u_k - \frac{\boldsymbol{\mu}_k}{r}. \qquad \text{(A2)}$$

For fixed $\boldsymbol{\xi}$ and $\boldsymbol{\mu}$, computing the optimal u leads to solving the linear PDE

$$-r(\Delta u - \text{div}(\boldsymbol{\xi}_{k+1})) + \text{div}(\boldsymbol{\mu}_k) + \lambda H^* H[u-f] = 0,$$

where H^* is the adjoint of the operator H. This can be done with the fast Fourier transform (FFT). Finally, the Lagrange multipliers are updated with

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k + \rho\left(\boldsymbol{\xi}_{k+1} - \nabla u_{k+1}\right).$$

Appendix B L_1 fitting term

Similarly, we rewrite (2) as

$$\int_\Omega a(\boldsymbol{x})|\boldsymbol{\xi}|^{p(\boldsymbol{x})} + \lambda\int_\Omega |r| \to \min$$

with the constraints

$$\boldsymbol{\xi} = \nabla u,$$

and

$$r = H[u-f].$$

The corresponding augmented Lagrangian is

$$L_{a,p}(u,r,\boldsymbol{\xi},\boldsymbol{\mu},z) = \lambda\int_\Omega |r| + \int_\Omega a(\boldsymbol{x})|\boldsymbol{\xi}|^{p(\boldsymbol{x})} \\ + \int_\Omega z(r - H[u-f]) \\ + \frac{\rho_1}{2}\int_\Omega |r - H[u-f]|^2 \\ + \int_\Omega \boldsymbol{\mu}\cdot(\boldsymbol{\xi}-\nabla u) + \frac{\rho_2}{2}\int_\Omega |\boldsymbol{\xi}-\nabla u|^2, \\ \text{(B3)}$$

where $\boldsymbol{\mu}(\boldsymbol{x})$ is vector of Lagrange multipliers associated with the constraint $\boldsymbol{\xi} = \nabla u$, $z(\boldsymbol{x})$ is the Lagrange multiplier associated with the constraint $r = H[u-f]$, and $\rho_1 > 0$, $\rho_2 > 0$ are constants. Similarly to above, we use the approximation

$$\boldsymbol{\xi}_{k+1} = \max\left\{|\boldsymbol{s}_k| - \frac{a(x)}{\rho_2|\nabla u_k|^{1-p(x)}}, 0\right\}\frac{\boldsymbol{s}_k}{|\boldsymbol{s}_k|},$$

where $\boldsymbol{s}_k = \nabla u_k - \frac{1}{\rho_2}\boldsymbol{\mu}_k$ for solving the sub-problem in $\boldsymbol{\xi}$. The L_1 fitting term gives a sub-problem in r which is solved by shrinkage

$$r_{k+1} = \max\left\{|H[u_k-f] - \frac{1}{\rho_1}z_k| - \frac{\lambda}{\rho_1}, 0\right\} \\ \text{sign}(H[u_k-f] - \frac{1}{\rho_1}z_k).$$

The updated u is obtained by solving the linear PDE

$$-\left(\rho_1 H^* H + \rho_2\Delta\right)u + \rho_1 H^* Hf - H^*(\rho_1 r - z)$$

$$-\operatorname{div}(\rho_2 \boldsymbol{\xi} - \boldsymbol{\mu}) = 0.$$

Finally, the Lagrange multipliers are updated with

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \rho_2(\boldsymbol{\xi}_{k+1} - \nabla u_{k+1})$$
$$z_{k+1} = z_k - \rho_1(r_{k+1} - H[u_{k+1} - f]).$$

References

1. Guo, X., Li, Y., Ling, H.: LIME: low-light image enhancement via illumination map estimation. IEEE Trans. Image Process. **26**(2), 982–993 (2017)
2. Ng, M.K., Ngan, H.Y., Yuan, X., Zhang, W.: Patterned fabric inspection and visualization by the method of image decomposition. IEEE Trans. Autom. Sci. Eng. **3**(11), 943–947 (2014)
3. Wakin, M., Romberg, J., Choi, H., Baraniuk, R.I.: Image compression using an efficient edge cartoon + texture model. In: Proceedings of the Data Compression Conference (DCC 2002), pp. 43–52 (2002)
4. Zhou, Y., Tang, Z., Hu, X.: Fast single image super resolution reconstruction via image separation. J. Netw. **9**(7), 1811–1818 (2014)
5. Xu, L., Yan, Q., Xia, Y., Jia, J.: Structure extraction from texture via relative total variation. ACM Trans. Graph. **31**(6), 139–113910 (2012)
6. Meyer, Y.: Oscillating patterns in image processing and nonlinear evolution equations. In: University Lecture Series, vol. 22. AMS, Boston (2001)
7. Aubert, G., Kornprobst, P.: Mathematical problems in image processing: partial differential equations and the calculus of variations, 2nd edn. In: Applied Mathematical Sciences, vol. 147. Springer, New York (2006)
8. Elad, M.: Sparse and Redundant Representations. Springer, New York (2010)
9. Vese, L.A., Le Guyader, C.: Variational Methods in Image Processing. CRC Press, New York (2016)
10. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**(1–4), 259–268 (1992)
11. Aujol, J.-F., Gilboa, G., Chan, T., Osher, S.: Structure-texture image decomposition: modeling, algorithms, and parameter selection. Int. J. Comput. Vis. **67**(1), 111–136 (2006)
12. Bergounioux, M.: Second order variational models for image texture analysis. Adv. Imaging Electron Phys. **181**, 35–124 (2014)
13. Jung, M., Kang, M.: Simultaneous cartoon and texture image restoration with higher-order regularization. SIAM J. Imaging Sci. **28**(1), 721–756 (2015)
14. Sun, Y., Schaefer, S., Wang, W.: Image structure retrieval via L_0 minimization. IEEE Trans. Vis. Comput. Graph. **24**(7), 2129–2139 (2017)
15. Ham, B., Cho, M., Ponce, J.: Robust guided image filtering using nonconvex potentials. IEEE Trans. Pattern Anal. Mach. Intell. **40**(1), 192–207 (2017)
16. Schaeffer, H., Osher, S.: A low patch-rank interpretation of texture. SIAM J. Imaging Sci. **6**(1), 226–262 (2013)
17. Ono, S., Miyata, T., Yamada, I.: Cartoon-texture image decomposition using blockwise low-rank texture characterization. IEEE Trans. Image Process. **23**(3), 1128–1142 (2014)
18. Gilles, J., Meyer, Y.: Properties of $bv - g$ structures + textures decomposition models. Application to road detection in satellite images. IEEE Trans. Image Process. **19**(11), 2793–2800 (2010)
19. Kim, Y., Ham, B., Do, M.N., Sohn, K.: Structure-texture image decomposition using deep variational priors. IEEE Trans. Image Process. **28**(6), 2692–2704 (2019). https://doi.org/10.1109/TIP.2018.2889531
20. Starck, J.-L., Elad, M., Donoho, D.L.: Image decomposition: separation of texture from piecewise smooth content. In: Optical Science and Technology, SPIE's 48th Annual Meeting, pp. 571–582. SPIE (2003)
21. Starck, J.-L., Elad, M., Donoho, D.L.: Image decomposition via the combination of sparse representations and a variational approach. IEEE Trans. Image Process. **14**(10), 1570–1582 (2005)
22. Peyré, G., Fadili, J., Starck, J.-L.: Learning adapted dictionaries for geometry and texture separation. In: Wavelets XII, vol. 6701. SPIE (2007)
23. Bobin, J., Starck, J.L., Fadili, J.M., Moudden, Y., Donoho, D.L.: Morphological component analysis: an adaptive thresholding strategy. IEEE Trans. Image Process. **16**(11), 2675–2681 (2007)
24. Peyré, G., Fadili, J., Starck, J.-L.: Learning the morphological diversity. SIAM J. Imaging Sci. **3**(3), 646–669 (2010)
25. Starck, J.L., Murtagh, F., Fadili, J.M.: Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity. Cambridge University Press, New York (2010)
26. Buades, A., Le, T.M., Morel, J.-M., Vese, L.A.: Fast cartoon + texture image filters. IEEE Trans. Image Process. **19**(8), 1978–1986 (2010)
27. Buades, A., Lisani, J.L.: Directional filters for color cartoon+ texture image and video decomposition. J. Math. Imaging Vis. **55**(1), 125–135 (2016)
28. He, K., Sun, J., Tang, X.: Guided image filtering. IEEE Trans. Pattern Anal. Mach. Intell. **35**(6), 1397–1409 (2012)
29. Kou, F., Chen, W., Wen, C., Li, Z.: Gradient domain guided image filtering. IEEE Trans. Image Process. **24**(11), 4528–4539 (2015). https://doi.org/10.1109/TIP.2015.2468183
30. Cho, H., Lee, H., Kang, H., Lee, S.: Bilateral texture filtering. ACM Trans. Graph. **33**(4), 128–11288 (2014)
31. Zhang, Q., Shen, X., Xu, L., Jia, J.: Rolling guidance filter. In: European Conference on Computer Vision (ECCV 2014), LNCS 8691, pp. 815–830 (2014)
32. Galetto, F.J., Deng, G., Al-Nasrawi, M., Waheed, W.: Edge-aware filter based on adaptive patch variance weighted average. IEEE Access **9**, 118291–118306 (2021). https://doi.org/10.1109/ACCESS.2021.3106907
33. Yang, Q.: Semantic filtering. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), pp. 4517–4526 (2016)
34. Zhang, H., Patel, V.M.: Convolutional sparse coding-based image decomposition. In: British Machine Vision Conference, BMVC, pp. 125–112511 (2016)
35. Papyan, V., Romano, Y., Sulam, J., Elad, M.: Convolutional dictionary learning via local processing. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5296–5304 (2017)
36. Xu, R., Xu, Y., Quan, Y., Ji, H.: Cartoon-texture image decomposition using orientation characteristics in patch recurrence. SIAM J. Imaging Sci. **13**(3), 1179–1210 (2020)
37. Xu, R., Xu, Y., Quan, Y.: Structure-texture image decomposition using discriminative patch recurrence. IEEE Trans. Image Process. **30**, 1542–1555 (2021). https://doi.org/10.1109/TIP.2020.3043665
38. Liu, W., Zhang, P., Lei, Y., Huang, X., Yang, J., Reid, I.: A generalized framework for edge-preserving and structure-preserving image smoothing. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11620–11628 (2020)
39. Liu, W., Zhang, P., Lei, Y., Huang, X., Yang, J., Ng, M.: A generalized framework for edge-preserving and structure-preserving image smoothing. Trans. Pattern Anal. Mach. Intell. **44**(10), 6631–6648 (2022)
40. Wang, L., He, D.-C.: Texture classification using texture spectrum. Pattern Recogn. **23**(8), 905–910 (1990)

41. Sur, F.: A non-local dual-domain approach to cartoon and texture decomposition. IEEE Trans. Image Process. **28**(4), 1882–1894 (2019)
42. Nixon, M.S., Aguado, A.S.: Feature Extraction & Image Processing for Computer Vision, 3rd edn. Academic Press, Orlando (2012)
43. Belyaev, A., Fayolle, P.-A.: Adaptive curvature-guided image filtering for structure+ texture image decomposition. IEEE Trans. Image Process. **27**(10), 5192–5203 (2018)
44. Fayolle, P.-A., Belyaev, A.G.: p-Laplace diffusion for distance function estimation, optimal transport approximation, and image enhancement. Comput. Aided Geom. Des. **67**, 1–20 (2018)
45. Diening, L., Harjulehto, P., Hästö, P., Ruzicka, M.: Lebesgue and Sobolev Spaces with Variable Exponents. Springer, Berlin (2011)
46. Radulescu, V.D., Repovs, D.D.: Partial Differential Equations with Variable Exponents: Variational Methods and Qualitative Analysis, vol. 9. CRC Press, New York (2015)
47. Tang, C., Hou, C., Hou, Y., Wang, P., Li, W.: An effective edge-preserving smoothing method for image manipulation. Digit. Signal Process. **63**, 10–24 (2017)
48. Elmoataz, A., Lezoray, O., Bougleux, S.: Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing. IEEE Trans. Image Process. **17**(7), 1047–1060 (2008)
49. Bougleux, S., Elmoataz, A., Melkemi, M.: Local and nonlocal discrete regularization on weighted graphs for image and mesh processing. Int. J. Comput. Vis. **84**(2), 220–236 (2009)
50. Gabay, D., Mercier, B.: A dual algorithm for the solution of nonlinear variational problems via finite element approximation. Comput. Math. Appl. **2**(1), 17–40 (1976)
51. Boyd, S., Parikh, N., Chu, E.: Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers. Now Publishers Inc, Boston (2011)
52. Kylberg, G.: The kylberg texture dataset v. 1.0. External report (Blue series) 35, Centre for Image Analysis, Swedish University of Agricultural Sciences and Uppsala University, Uppsala, Sweden (September 2011)
53. Liu, C., Yang, C., Wei, M., Wang, J.: Texture smoothing quality assessment via information entropy. IEEE Access **8**, 88410–88421 (2020). https://doi.org/10.1109/ACCESS.2020.2993146

Pierre-Alain Fayolle is an associate professor at the University of Aizu. His research interests include shape and solid modeling, computational science, and applied mathematics

Alexander Belyaev is an associate professor at School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, UK. His research interests include digital geometry processing, mathematical image analysis, applied partial differential equations, and recreational mathematics.