Hierarchical morphological graph signal multi-layer decomposition for editing applications

Olivier Lézoray

IET Journals
The Institution of Engineering and Technology

Hierarchical morphological graph signal multi-layer decomposition for editing applications

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Olivier Lézoray[1]

[1] *Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France*
** E-mail: olivier.lezoray@unicaen.fr*

Abstract: We address the problem of editing signals such as 2D color images or 3D colored meshes that are represented under the general framework of graph signals. As state-of-the-art editing approaches decompose an image into several layers in order to manipulate them, we propose a hierarchical multi-layer decomposition of graph signals that relies on morphological filtering. Since morphological filtering operators require a complete lattice, a dedicated approach for the morphological processing of vectorial data on graphs is used. By iterating the application of morphological filterings of decreasing sizes, the graph signal is decomposed into several detail layers, each capturing a given detail level. Editing applications such as abstraction, sharpness enhancement and tone mapping are shown to illustrate the benefits of the proposed approach.

1 Introduction

Image editing refers to the operations by which the content of an image is manipulated to improve its aesthetical appeal or its visual quality. There are a lot of possible ways to visually edit images with computational techniques, e.g., abstraction, style transfer, removal of unwanted objects, sharpening, compositing, matting, to quote a few. As social media platforms such as Instagram have rapidly grown, photographs are now ubiquitous. This has created a demand for image editing applications that enable untrained users to enhance and manipulate photos. However, developing computationally efficient approaches to produce artifact-free subjective and qualitative enhancements is difficult. State-of-the-art approaches for image editing have adopted the approach of Farbman *et al.* [1] that has proposed to decompose an image into a base layer and several detail layers, each capturing a given scale of details. This principle has also been explored in other image processing works [2–5]. The image decomposition aim is to break down an image into both coarse structures and fine details. The obtained decomposition can then facilitate the subsequent manipulation of the image for computational image editing tasks. To perform the decomposition, state-of-the-art approaches have considered structure-preserving image smoothing methods. They enable to eliminate unimportant fine-scale details while maintaining the major image structures, such as salient edges and contours. Over the past decade, many structure preserving smoothing techniques have been proposed [6–9]. Some works rely on filtering operating within a local window. Typical approaches are the Bilateral filter (BLF) [10] and the Rolling Guidance Filter (RGF) [6]. Some works rely on global optimization. The smoothed image is obtained by solving a global objective function involving a data term and a regularization term. Typical approaches are the Fast Global Smoother (FGS) [11], the L_0 [9] or L_1 smoothing [12]. Beyond these recent approaches, structure-preserving filters have a long history, the review of which is beyond the scope of this paper. Such works have considered different formalisms such as anisotropic diffusion [13], PDEs [14], Empirical Mode Decomposition [15–18], sparse representations [19, 20], etc [3, 5]. Whatever their formulation, structure-preserving filters are the key element used in image editing tasks such as detail manipulation or visual abstraction. The design of structure preserving filters for editing tasks still is a hot topic since there are many issues that can occur. Halos can appear around edges due to over-smoothing. Gradient reversals can also appear around edges if the base layer is not consistent with the original image. Balancing the trade-off between preserving structures and suppressing texture is not a trivial task.

In this paper, we propose to investigate the use of mathematical morphology (denoted as MM in the sequel) operators for computational editing tasks. Contrary to the state-of-the-art approaches that consider only images, we also consider the challenging case of 3D meshes [21]. This is tackled within the formalism of graph signals that enables to process in a unified manner any type of signal on a graph. As said before, we will consider MM filtering operators. The latter are filters that preserve input space values (i.e., no new value is introduced), and are therefore by definition not subject to the production of halos, which is a common problem met in computational image editing tasks [1]. So far very few approaches have considered morphological operators for image editing. Subr *et al.* have proposed [22] to consider the local extrema of images to extract details and have combined ideas from Empirical Mode Decomposition (EMD) [23] and morphological filtering [24]. As explained by Subr *et al.* [22], EMD approaches [17, 18] have been little considered in image editing as they do not well preserve edges and similarly for morphological approaches as they do not well preserve shapes. To overcome this, some works have designed adaptive morphological filters [25–28] or proposed levelings [29–31] that can better preserve shapes. Nevertheless, morphological approaches have many benefits. First, they enable a computationally efficient nonlinear processing that can furthermore be interpreted in terms of partial differential equations [32, 33]. Second, the core erosion and dilation operators can be used to form more complex operations such as closings or openings. For instance, Alternate Sequential Filters (ASF) can be defined from the concatenation of openings and closings with a progressively increasing structuring element, that has been proven effective for multi-scale analysis of shape [34]. In this paper we consider such a morphological operator (the Open Close Close Open [35]) to obtain a decomposition of graph signals.

However, the use of algebraic morphological operators is based on the theory of complete lattices [36] that requires a total (or a partial) ordering between all the elements to be processed. Total orders are generally preferable to avoid the appearance of false colors (particularly problematic for editing applications), even if, as mentioned by Chevallier and Angulo [37], they can introduce irregularity issues. While MM is well defined for scalars, there is no generally accepted way of performing morphological operations on arbitrary vectors since there is no natural order on the latter. We have recently proposed [38] a framework for the construction of complete lattices for any kind of vector data. The principle consists in learning the complete lattice structure from a modeling of the variety on which the vectors do live. In this paper, this complete lattice learning framework is considered for the morphological editing of general graph signals. The formalism of graph signals [39] is considered to allow an easy application of the proposed framework to both colored images and 3D meshes. Then we propose a hierarchical multi-layer decomposition of graph signals that relies on morphological filtering. This decomposition can be manipulated for computational editing applications such as sharpness enhancement and tone mapping.

The outline of the paper is a follows: in section 2 we detail how to build a complete lattice by learning the vectors manifold and how this provides a graph signal representation that enables the morphological processing of multivariate graph signals; from this, the construction of a hierarchical decomposition of a graph signal into layers is introduced; section 4 presents computational editing examples with the proposed approach; last section is the conclusion.

2 Manifold-based complete lattice learning

2.1 Notations

A graph is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{v_1, \ldots, v_m\}$ is a finite set of vertices and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ a finite set of edges. An edge of $\mathcal{E}$ connects two vertices v_i and v_j of $\mathcal{V}$ and is denoted by (v_i, v_j). Such two vertices are called two adjacent vertices and the notation $v_i \sim v_j$ is used to describe this. Only un-directed and un-weighted graphs will be considered in this paper. Indeed, since we will process signals on either regular domains (images represented as grid graphs) or triangulated domains (3D data represented as meshes), the graphs are known beforehand and are undirected and unweighted. A function that associates a vector to each vertex of the graph is called a graph signal. This corresponds to the mapping $f : \mathcal{G} \to \mathcal{F} \subset \mathbb{R}^n$ where $\mathcal{F}$ is a non-empty set of vectors. Therefore, given a graph, a vector $\mathbf{v}_i = f(v_i)$ is associated to each vertex $v_i \in \mathcal{G}$. The (multi)-set $\mathcal{T} = \{\{\mathbf{v}_1, \cdots, \mathbf{v}_m\}\}$ corresponds to all the vectors that are associated to the vertices of the graph. Consequently, one has $|\mathcal{V}| = m$. For the sake of simplicity, we consider that each vertex v_i of the graph has a specific associated vector $\mathbf{v}_i$ even if $\exists v_j | f(v_j) = f(v_i)$. Therefore, there can be duplicated vectors in $\mathcal{T}$ and $\mathcal{F} \subset \mathcal{T}$. The i-th element of this set will be denoted by $\mathcal{T}[i] = \mathbf{v}_i$.

2.2 Complete lattices

Given the vertices vectors $\mathcal{T}$, a vectorial ordering of the vectors of the set $\mathcal{T}$ is mandatory to be able to perform morphological operations on graph signals. This ordering relationship between the vectors is related to the definition of a complete lattice $(\mathcal{T}, \leq)$, a key item for the definition of mathematical morphology operators [36]. As mentioned before, there is no generally accepted way of ordering vectors [40]. For specific types of vectors, such as colors, specific orderings have been proposed. Most of them use lexicographic orderings [41, 42] with variations on the ordering of channels (often with a color space change). Color images can also be represented as a tensor field in the HSL color space and one can consider the Loewner ordering to compare the symmetric matrices and perform morphological processing [43, 44]. A more general and convenient way to define an ordering relation between the vectors of a set $\mathcal{T}$ is to use the framework of h-orderings [45]. This has been considered in some recent works [38, 46, 47]. The principle of this approach consists in defining a surjective transform h from $\mathcal{T}$ to $\mathcal{L}$. The set $\mathcal{L}$ has to be a complete lattice that can be equipped with the conditional total ordering (i.e., the lexicographic ordering) [45]. We refer to $\leq_h$ as the h-ordering given by:

$$h : \mathcal{T} \to \mathcal{L} \text{ and } \mathbf{v} \to h(\mathbf{v}), \forall (\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{T} \times \mathcal{T}$$

$$\mathbf{v}_i \leq_h \mathbf{v}_j \Leftrightarrow h(\mathbf{v}_i) \leq h(\mathbf{v}_j) \ . \tag{1}$$

The advantage is that the ordering of the set $\mathcal{T}$ can be deduced directly from $\mathcal{L}$ by means of h [48], and consequently $\mathcal{T}$ is no longer required to be a complete lattice. Moreover, if the projection h is bijective, equivalences with space filling curves [49] and rank transforms [50] can be exhibited. However, the projection h cannot be linear since a distortion of the initial vector space is unavoidable [37, 51] (space-filling curves topological preservation arguments can be used to prove this [52]). To cope with this, we consider a nonlinear approach for the construction of the mapping h and then deduce the complete lattice $(\mathcal{T}, \leq_h)$.

2.3 Manifold-based complete lattice

In this section, we recall the basic ingredients of the approach we have recently proposed [38] to learn a complete lattice.

2.3.1 Complete lattice learning:

The principle of our approach is to model the manifold of the vectors of a graph signal and to define the ordering of the vectors from the representation of this manifold. To model the latter, we consider a non-linear dimensionality reduction technique: Laplacian EigenMaps [53]. However, this technique requires to perform the eigen-decomposition of the normalized Laplacian which is very computationally demanding. To cope with this, we use a dictionary to represent the data. This dictionary being small, Laplacian EigenMaps can be performed on it. The obtained projection operator $h_{\mathcal{D}}$ corresponds to construct a $h_{\mathcal{D}}$-ordering from the dictionary $\mathcal{D}$ and a new representation $h_{\mathcal{D}}(\mathbf{x}'_i)$ is obtained for each element $\mathbf{x}'_i$ of $\mathcal{D}$. Since the obtained representation is valid only for the dictionary, it is interpolated to all the vectors of the initial lattice $\mathcal{T}$ with the Nyström extrapolation [54]. Algorithm 1 provides the outline of this complete lattice learning.

Algorithm 1 Learning a complete lattice from a manifold of vectors

A graph signal f corresponds to a set $\mathcal{T}$ of m vectors in $\mathbb{R}^n$.
A dictionary $\mathcal{D} = \{\mathbf{x}'_1, \cdots, \mathbf{x}'_p\}$ of $p \ll m$ vectors is built by Vector Quantization [55].
The similarity matrix $\mathbf{K}_{\mathcal{D}}$ between the vectors $\mathbf{x}'_i \in \mathcal{D}$ is computed with

$$K_{\mathcal{D}}(i,j) = \exp\left(-\frac{\|\mathbf{x}'_i - \mathbf{x}'_j\|_2^2}{\sigma^2}\right)$$

and $\sigma = \max_{(\mathbf{x}'_i, \mathbf{x}'_j) \in \mathcal{D}} \|\mathbf{x}'_i - \mathbf{x}'_j\|_2^2$
Compute $\mathbf{D}_{\mathcal{D}}$ the degree diagonal matrix of $\mathbf{K}_{\mathcal{D}}$
Compute the normalized Laplacian $\mathbf{L} = \mathbf{I} - \mathbf{D}_{\mathcal{D}}^{-\frac{1}{2}} \mathbf{K}_{\mathcal{D}} \mathbf{D}_{\mathcal{D}}^{-\frac{1}{2}}$
The eigen-decomposition of $\mathbf{L}$ provides eigenvectors $\mathbf{\Phi}_{\mathcal{D}} = [\mathbf{\Phi}^1_{\mathcal{D}}, \cdots, \mathbf{\Phi}^p_{\mathcal{D}}]$ and eigenvalues $\mathbf{\Pi}_{\mathcal{D}} = \mathrm{diag}[\lambda_1, \cdots, \lambda_p]$.
Compute the similarity matrix $\mathbf{K}_{\mathcal{DT}}$ between sets $\mathcal{D}$ and $\mathcal{T}$ and the associated degree diagonal matrix $\mathbf{D}_{\mathcal{DT}}$
Extrapolate the eigenvectors of $\mathcal{D}$ to $\mathcal{T}$ with

$$\tilde{\mathbf{\Phi}} = \mathbf{D}_{\mathcal{DT}}^{-\frac{1}{2}} \mathbf{K}_{\mathcal{DT}}^T \mathbf{D}_{\mathcal{D}}^{-\frac{1}{2}} \mathbf{\Phi}_{\mathcal{D}} (\mathrm{diag}[\mathbb{1}] - \mathbf{\Pi}_{\mathcal{D}})^{-1}$$

Finally, the projection operator $h : \mathcal{T} \subset \mathbb{R}^n \to \mathcal{L} \subset \mathbb{R}^p$ is given by $h(\mathbf{x}) = (\tilde{\phi}^1(\mathbf{x}), \cdots, \tilde{\phi}^p(\mathbf{x}))^T$ and the manifold-based learned complete lattice is defined by $(\mathcal{T}, \leq_h)$. What Figure 1 shows for two given graph signals is: the learned dictionaries, the learned manifolds from the dictionaries and the final manifolds by Nyström extrapolation.

2.3.2 Graph signal representation:

Once the complete lattice $(\mathcal{T}, \leq_h)$ is available, a new graph signal representation can be defined from it. Indeed, from the manifold-based ordering $\leq_h$, all the

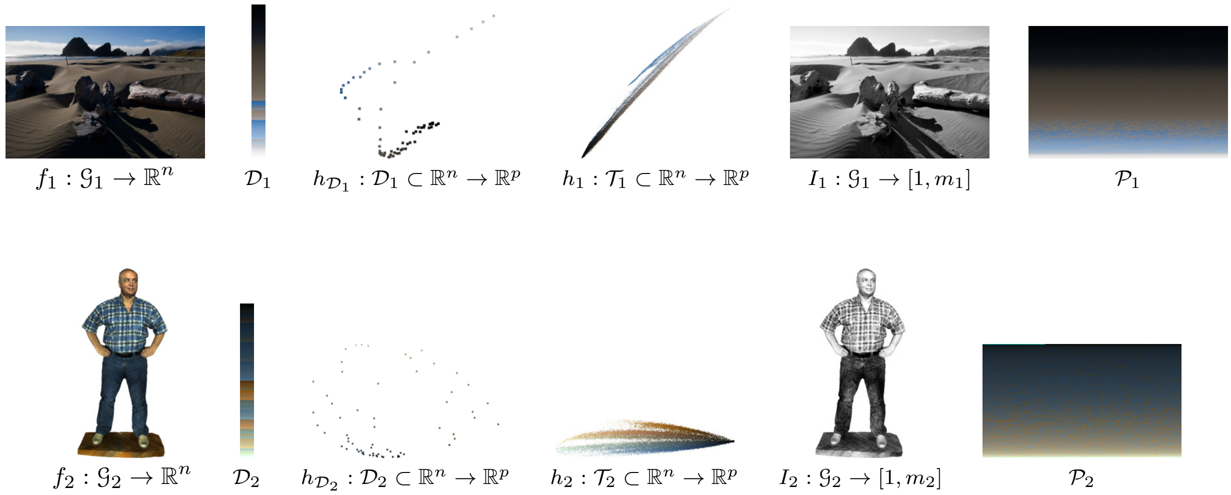


Fig. 1: Illustration showing the different steps of the manifold-based complete lattice learning. This is shown for two different graph signals: colors on a grid graph (first line) and on a 3D mesh (second line). Given an original graph signal f_i ($i = 1$ for the image, and $i = 2$ for the 3D colored mesh), a dictionary $\mathcal{D}_i$ is first constructed and p representative color vectors are obtained. A mapping $h_{\mathcal{D}_i}$ is then obtained by manifold learning on the dictionary $\mathcal{D}_i$. This mapping is presented here by showing the dictionary colors on the three first eigenvectors of the projection. The mapping $h_{\mathcal{D}_i}$ is then interpolated to all the colors of the original signal and the final mapping h_i is obtained (also presented by showing the colors of the signal on the three first eigenvectors of the interpolated projection). Finally, the index graph signals I_i and the sorted vectors $\mathcal{P}_i$ obtained from the induced manifold-based ordering $\leq_{h_i}$ are presented.

vectors of $\mathcal{T}$ can be sorted and a sorted permutation $\mathcal{P}$ is obtained:

$$\mathcal{P} = \{\mathbf{v}'_1, \cdots, \mathbf{v}'_m\} \text{ with } \mathbf{v}'_i \leq_h \mathbf{v}'_{i+1}, \forall i \in [1, (m-1)]. \quad (2)$$

Given this ordering of the set $\mathcal{T}$, a new graph signal in the form of an index can be defined. We denote by $I : \mathcal{G} \to [1, m]$ this index graph signal. It is defined by:

$$I(v_i) = \{k \mid \mathbf{v}'_k = f(v_i) = \mathbf{v}_i\} \ . \quad (3)$$

Therefore, the index graph signal associates to each vertex v_i of the graph the rank of its corresponding vector $f(v_i)$ in $\mathcal{P}$, the set of sorted vectors. The pair $(I, \mathcal{P})$ defines a new representation of the graph signal f denoted as f^*. The first element of the pair, I, is called the index. The second element of the pair, $\mathcal{P}$, is called the palette. Recovering the original graph signal f can be easily done since $f(v_i) = \mathcal{P}[I(v_i)] = \mathcal{T}[i] = \mathbf{v}_i$. Figure 1 shows examples of obtained graph signal representations for two graph signals on two different graphs.

3 Morphological hierarchical graph signal multi-layer decomposition

3.1 Graph signal morphological processing

Now we present how to perform morphological processing operations with the new representation $(I, \mathcal{P})$ of a graph signal f. The index graph signal I is a scalar signal with values in the range $[1, m]$. This index graph signal can be processed with any processing operation and a processed index graph signal I' is obtained. However, the reconstruction of the final processed graph signal f' from a processed index graph signal I' is not possible for any kind of processing. Indeed, reconstructing f' from $(I', \mathcal{P})$ requires that the values of I' are kept within the integer range $[1, m]$ to be able to associate a vector of the manifold-based ordering $\mathcal{P}$ to each value of I'. This means that the processing that can be performed on an index graph signal must be a vector-preserving signal processing method: no new values can be created. Morphological operations are vector preserving methods and this is the reason why we have considered them in this paper. For a specific morphological processing g applied to the index graph signal, the corresponding processed graph signal can be obtained by

$$g(f(v_i)) = \mathcal{P}[g(I(v_i))] \ . \quad (4)$$

For a structuring element $B_k \subset \mathcal{G}$, the erosion and dilation of a graph signal f at vertex $v_i \in \mathcal{G}$ are defined as:

$$\epsilon_{B_k}(f)(v_i) = \{\mathcal{P}[\wedge I(v_j)], v_j \in B_k(v_i)\} \quad (5)$$

$$\delta_{B_k}(f)(v_i) = \{\mathcal{P}[\vee I(v_j)], v_j \in B_k(v_i)\} \ . \quad (6)$$

A structuring element $B_k(v_i)$ is a subgraph of the processed graph. Given a vertex v_i and a radius size k, the structuring element is composed of all the vertices that can be reached from v_i in k walks:

$$B_k(v_i) = \begin{cases} \{v_j \sim v_i\} \cup \{v_i\} \text{ if } k = 1 \\ B_{k-1}(v_i) \cup \left(\cup_{\forall v_l \in B_{k-1}(v_i)} B_1(v_l) \right) \text{ if } k \geq 2 \end{cases} \quad (7)$$

The number of vertices in a given k-hop neighborhood $B_k(v_i)$ depends on the vertex v_i when the graph is irregular. However, the associated erosion and dilation are symmetry preserving operators [56]. From the basic erosion and dilation operators, we can derive all the standard morphological filters for graph signals such as openings $\gamma_{B_k}(f) = \delta_{B_k}(\epsilon_{B_k}(f))$ and closings $\phi_{B_k}(f) = \epsilon_{B_k}(\delta_{B_k}(f))$. Figures 2 and 3 present examples of morphological filtering of a 2D color image and a 3D colored mesh. As it can be seen, the morphological filters can be useful for simplifying a graph signal in a coarse manner with a large structuring element.

3.2 Graph signal multi-layer decomposition

State-of-the-art approaches for image editing have adopted the approach of Farbman *et al.* [1] that has proposed to decompose an image into a base layer and several detail layers, each capturing a given scale of details. This approach takes its roots from image decomposition works [2] the aim of which was to decompose an image into two components. The first component contains geometrical information, and the second contains oscillations, i.e., noise and textures. By iterating this approach on the oscillating part, one can get a hierarchical decomposition [5, 16]. We consider a similar approach but consider the general case of graph signals and make use of morphological filtering to decompose a graph signal into l layers,

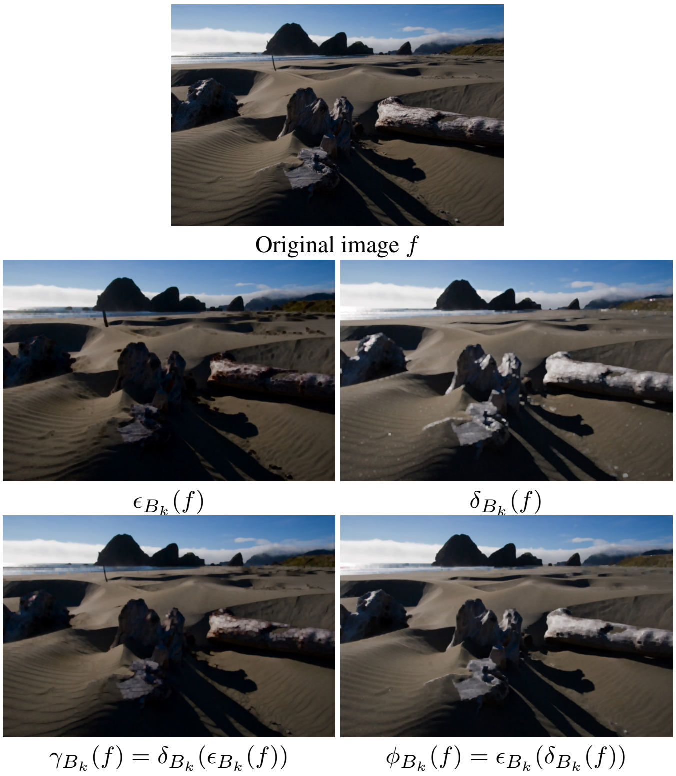


Fig. 2: From top to bottom, left to right: examples of morphological processing of a color image (grid-graph signal) with $k = 5$.

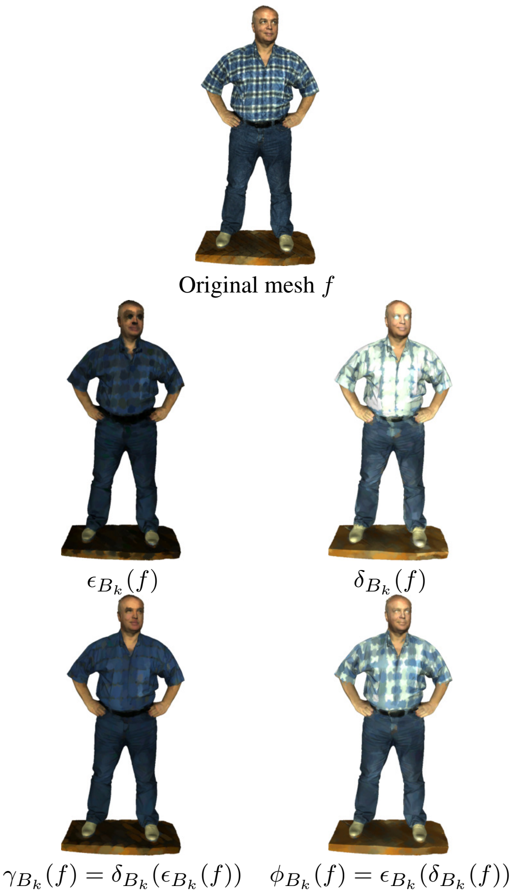


Fig. 3: From top to bottom, left to right: examples of morphological processing of a 3D colored mesh with $k = 5$.

Algorithm 2 Hierarchical multi-layer morphological decomposition of a graph signal

$d_{-1} = f, i = 0$
while $i < l$ **do**
  Computation of the graph signal representation at level $i - 1$ from d_{i-1}: $d^*_{i-1} = (I_{i-1}, \mathcal{P}_{i-1})$
  Morphological Filtering of d^*_{i-1}:
  $f_i = MF_{B_{l-i}}(d^*_{i-1})$
  Computation of the residual (detail layer):
  $d_i = d_{i-1} - f_i$
  Continue to the next layer:
  $i = i + 1$
end while

as shown in Algorithm 2. This leads to the development of a hierarchical morphological graph signal multi-layer decomposition. Such a decomposition of a graph signal into layers ranging from coarse to fine details imposes constraints on the decomposition. The first layer f_0 must correspond to the coarsest layer of the graph signal, whereas the other layers f_i must correspond to details that are finer as one progresses into the decomposition levels. In our proposed hierarchical morphological decomposition, this imposes that the scale of filtering is decreasing along the layers. In terms of morphological filtering, this means that the size of the structuring element used in the morphological filtering (MF) has to decrease along the layers. This can be seen in Algorithm 2 where the structuring element B_{l-i} has a size that depends on the number of levels of decomposition and that is decreasing along the levels (with $i \in [0, l-1]$). By proceeding this way, it is ensured that each level captures a specific amount of detail corresponding to the structuring element size and the successive layers will extract details that will become finer [35, 57]. In addition, each detail layer d_i is the residual between the previous layer d_{i-1} and its morphological processing. Therefore, the set of vectors that constitute graph signal d_i is different from the set of vectors that constitute the previous layer d_{i-1}. This is why a graph signal representation $(I_i, \mathcal{P}_i)$ has to be computed for each of the successive layers to be decomposed. At the end of the decomposition, the original graph signal is represented as the sum of several layers:

$$f = \sum_{i=0}^{l-2} f_i + d_{l-1} \quad . \tag{8}$$

The f_i's do represent the different decomposition layers of f that capture details at different scales (from coarse to fine). One thing remains to be fixed in Algorithm 2: what morphological filtering to use ? Many choices are possible and we have chosen an Open Close Close Open (OCCO) filter. This OCCO filter is a self-dual operator that has been shown to have excellent signal decomposition abilities [35]:

$$OCCO_{B_k}(f) = \left\lfloor \frac{\gamma_{B_k}(\phi_{B_k}(f)) + \phi_{B_k}(\gamma_{B_k}(f))}{2} \right\rfloor \quad . \tag{9}$$

In Figures 4 and 5, we show two examples for two graph signals (a color image and a 3D colored mesh) with five levels of decomposition ($l = 5$) to obtain a coarse base layer f_0, two medium detail layers f_1 and f_2, and two fine detail layers f_3 and d_3 such as $f = f_0 + f_1 + f_2 + f_3 + d_3$. These layers can be manipulated for editing applications and we investigate this in the next section.

3.3 *Computational complexity analysis*

We analyze the computational complexity of our approach with respect to state-of-the-art approaches. Our approach performs a hierarchical decomposition into l levels defined recursively using residual images. Each level involves the computation of a graph signal representation (in the form of a palette and an index) with the morphological filtering of the index. The complexity of the graph signal representation computation is $O(pm)$ with m the number of pixels and $p \ll m$ the size of the dictionary. The complexity of the filtering is $O(2 \times k \times m)$ since each morphological operator for erosion or dilation has a complexity of $O(k \times m)$ with k the size of the structuring element. The complexity of the whole decomposition is therefore $O(l \times (p + 2k) \times m)$. State-of-the art approaches only require a filtering the complexity of which usually is $O(n \times k \times m)$ with n the number of iterations of the filtering. The complexity of the induced whole decomposition is therefore $O(l \times n \times k \times m)$. As can be seen, our approach is competitive with these ones only if $(p + 2k) \leq n \times k$ which implies to consider only dictionaries of relatively small sizes.

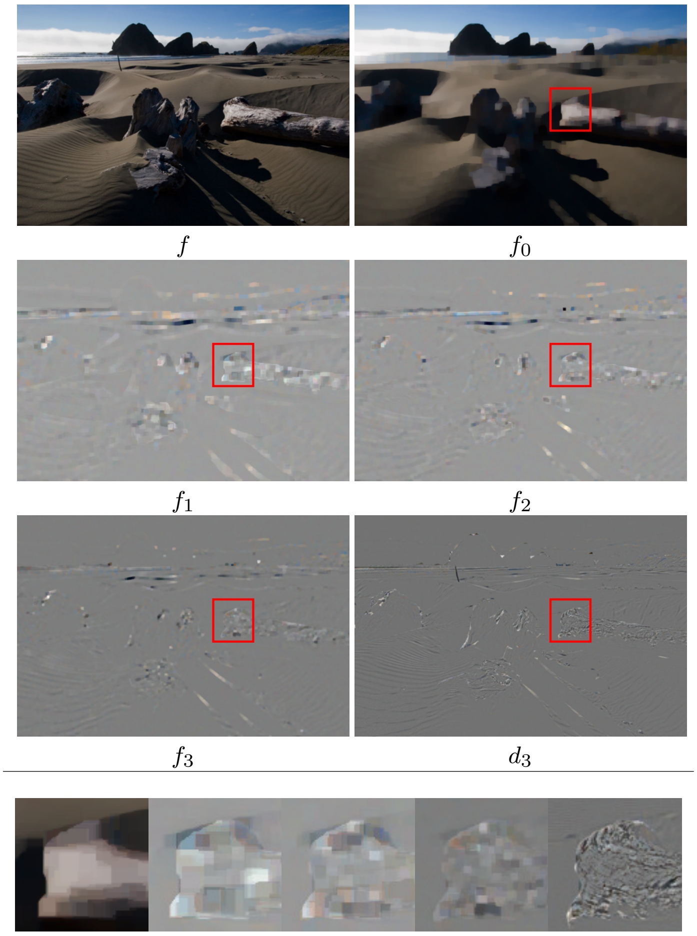


Fig. 4: First part: from top to bottom, left to right: an original image f, and its decomposition into five layers f_0, f_1, f_2, f_3,and d_3. Second part: cropped and zoomed part of the layers to show the increasing level of detail extraction along the layers.

4 Graph signal editing applications

In this section we provide several examples of graph signal editing applications that make use of the proposed hierarchical morphological multi-layer decomposition. To show the benefit of using the formalism of graph signals, we will consider two different types of graph signals [58]. The first graph signal type corresponds to 2D color images that are modeled as color vectors assigned to the vertices of a 8-adjacency regular grid graph. The second graph signal type corresponds to 3D colored meshes that are modeled as color vectors assigned to the vertices of a 3D mesh (it is an irregular graph). This last type of graph signal is much less common than ordinary 2D color images. It is now possible to obtain graph signal of this type with specific 3D color scanners that enable to simultaneously acquire the spectral (R, G, B colors) and spatial (X, Y, Z coordinates) values of the points of an object. Very few works have addressed the problem of editing such a graph signal type [57, 59, 60].

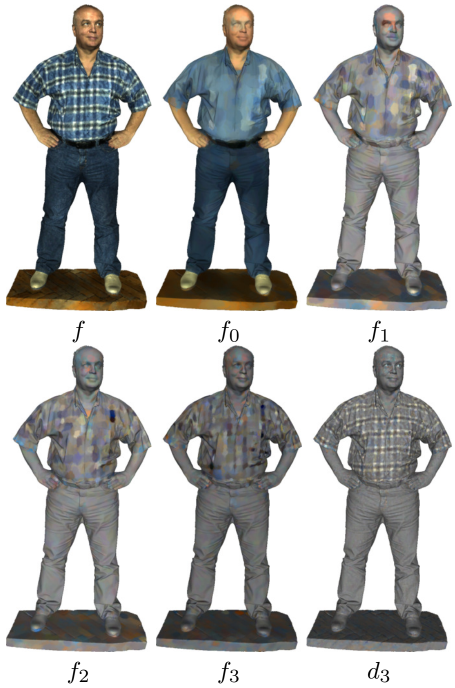


Fig. 5: From top to bottom, left to right: an original mesh f, and its decomposition into five layers f_0, f_1, f_2, f_3,and d_3.

4.1 Abstraction

First, we consider the typical application of smoothing filters: producing a simplified version of the image to generate an abstract illustration of it. The goal of image smoothing is therefore to eliminate unimportant fine-scale details while maintaining primary image structures. With our approach, named MF (for Morphological Filtering), the image is decomposed in two layers, and the first layer is the result of image smoothing. Figures 6 and 7 compare the smoothing results of our method to ten existing methods: SUL [61], SGF [62], SDF [63], L_1 [12], BTLF [7], FGS [11], RGF [6], RTV [64], L_0 [9], BLF [10]. In these figures, the results were obtained by considering only the first layer f_0 of the decomposition. Figures 6 and 7 also show the resultant brightness values on a 1D image slice indicated by the red line on the original image: the curve should be flattened while preserving the important transitions. As can be seen, our approach produces visual results competitive with the state-of-the-art, close to the results of RGF [6], RTV [64] and BTLF [7]. With our approach, most of the high contrast structures have been preserved and fine scale details (e.g., the rope of the paraglider) have been removed. Moreover, flat regions are obtained in areas where similar colors do appear but the important object edges are preserved without any blurring. As previously mentioned, this is a property that is critical for any structure-preserving filtering. However, as in state-of-the-art literature, this comparison is only visual. Indeed, up to now, there is no established method or dataset to quantitatively evaluate the performance of image smoothing methods. In our experiments, in order to ease the comparison with other approaches, we consider several quality measures to compare the results: two full-reference quality metrics PSNR (Peak Signal to Noise Ratio [65]) and SSIM (Structural SIMilarity [66]), and one no-reference quality metric BRISQUE (Blind/Referenceless Image Spatial Quality Evaluator [67]). For all these metrics, we have normalized their values between 0 and 1. The PSNR estimates the fidelity to the original image (the higher the better). The SSIM estimates the perceived change in structural information with respect to the original image (the higher the better).

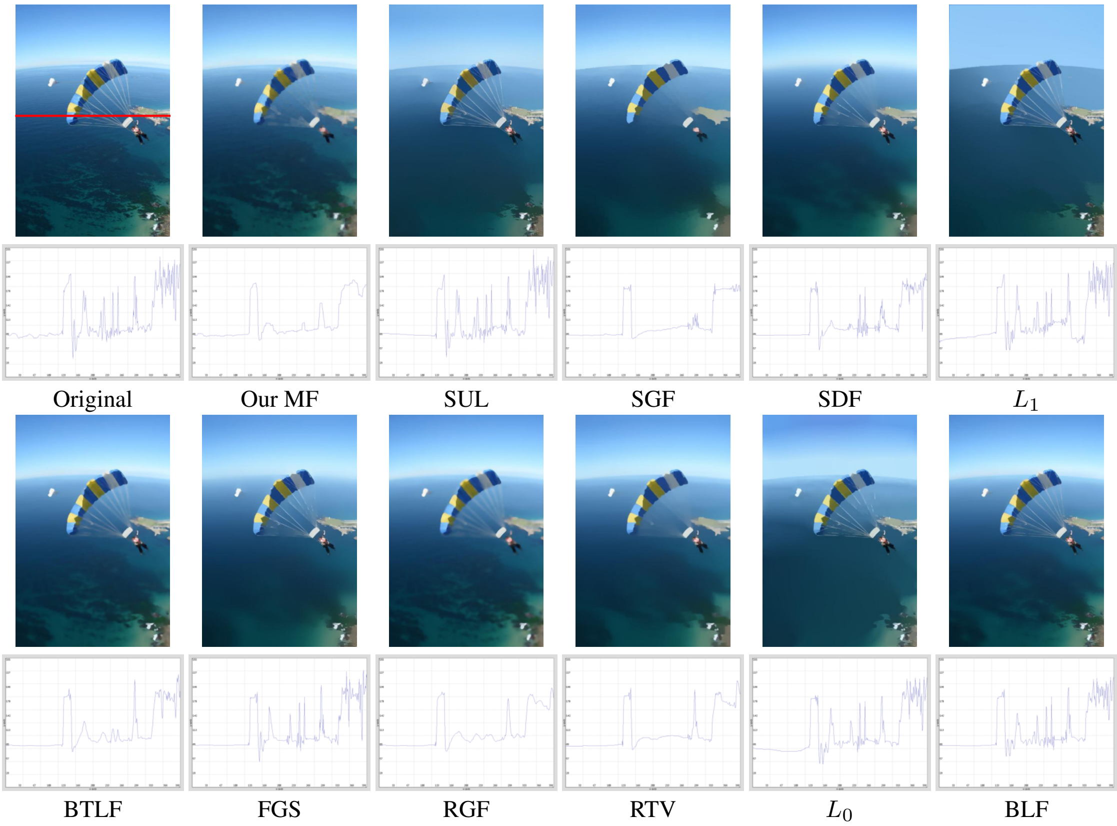


Fig. 6: Visual comparison on the Paragliding image between our method (named MF) and previous image smoothing methods, abbreviated as SUL [61], SGF [62], SDF [63], L_1 [12], BTLF [7], FGS [11], RGF [6], RTV [64], L_0 [9], BLF [10]. The brightness values on a 1D image slice (indicated by the red line on the original image) are shown under each image.

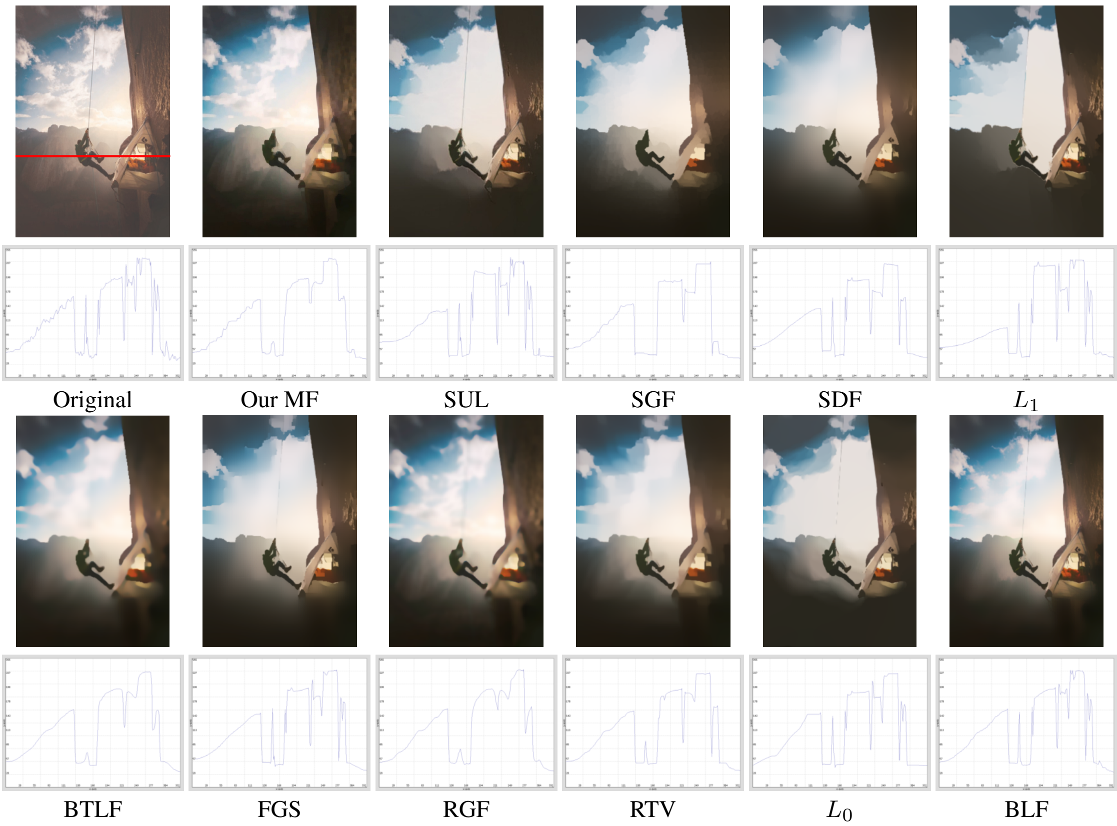


Fig. 7: Visual comparison on the Climber image between our method (named MF) and previous image smoothing methods, abbreviated as SUL [61], SGF [62], SDF [63], L_1 [12], BTLF [7], FGS [11], RGF [6], RTV [64], L_0 [9], BLF [10]. The brightness values on a 1D image slice (indicated by the red line on the original image) are shown under each image.

The BRISQUE exploits spatial natural scene statistics to assess the visual quality of an image without the need of the original image (the lower the better). Each quality indicator measures an objective that an image abstraction should attain: colors close to the original (assessed by the PSNR), the main edges should be preserved (assessed by the SSIM), and the global result should be of good visual quality (assessed by the BRISQUE). Since an abstraction approach should fulfill all these requirements, we also consider the value SSIM*PSNR/BRISQUE as a global quality indicator. Figure 9 presents the results for the considered images (called Climber for Figure 7 and Paragliding and for Figure 6). As it can be seen, our approach is competitive with all the other approaches with respect to all the metrics and is among the fourth best ones with respect to the global quality indicator. Figure 8 presents smoothing results for 3D colored meshes with a comparison with graph TV filtering [4, 57] (the sole other state-of-the-art approach to compare with). As for images, our approach can produce pleasant smoothing results with less blurry effect and a more piecewise constant result. In particular, the fire hydrant 3D colored mesh has been obtained using photogrammetry and the stitching defects appearing have been suppressed by our approach (see the zoomed parts in Figure 8). Again, to complement these visual results, we consider quality indicators (different from the ones for images, since SSIM and BRISQUE are defined only for images). We provide the values of the PSNR, of the smoothing level (the ratio $\|f - f_0\|/\|f\|$, the higher the smoother) and the product of both as a global quality indicator. Figure 9 shows these results for the four considered meshes (called Street, Vase, Firehydrant and AltarPiece). The results are very close for the meshes Vase and AltarPiece and for the meshes Street and Fire-Hydrant the global quality is in favor of our approach. To assess the blurry effect, we consider the Tenengrad criterion [68, 69]. It has been shown [69] that a high value of this criterion means a sharper signal and that this value is correlated with perceived sharpness. The values are shown in Figure 8 and are clearly in benefit of our approach. All these results show the interest of morphological operators for abstraction editing applications.

A smoothed graph signal can also be used to produce a non-photorealistic abstraction. The principle consists in: extracting edges from the smoothed image, emphasizing them, adding them back to the smoothed graph signal to enhance the differences between the different regions composing the image. Figure 10 presents two examples of such non-photorealistic abstraction results along with a comparison of the result obtained with SUL [61]. As can be seen, our result is visually much less blurry than SUL [61]. This is confirmed by the Tenengrad sharpness measure [69] (the higher, the sharper).

4.2 *Detail manipulation*

The second typical application of smoothing filters we consider is detail magnification, which is also commonly called sharpness enhancement. Their aim is to manipulate the signal to sharpen its prominent edges. Existing techniques for sharpness enhancement of images use structure-preserving smoothing filters [6–9] within a hierarchical decomposition framework. The image is decomposed into several layers containing from coarse to fine scale details. The layers are then individually manipulated (by boosting them by a multiplicative factor) to perform detail manipulation and enhancement. Similar approaches have been proposed for the specific case of 3D mesh vertices coordinates manipulation [21] or 3D colored mesh color manipulation [60]. With our approach, the multi-layer morphological decomposition of the graph signal gradually decomposes an input graph signal into coarse to fine scales layers. These layers are then manipulated by non-linear boosting curves and blended together with the use of a structure mask to produce an enhanced graph signal. We detail this in the sequel.

4.2.1 Proposed approach: For a given graph signal $f = (\mathcal{I}, \mathcal{P})$, its multi-layer decomposition into l levels is constructed first. The obtained layers are manipulated individually with specific boosting coefficients. The manipulated layers are then added

Fig. 8: Visual comparison between our method (named MF) and another 3D colored mesh smoothing method, abbreviated as GTV [57]. For each mesh (called Street, Vase, Firehydrant and Altar-Piece), two lines present the results. First line shows original and filtered versions. Second line presents zoomed parts of the meshes to show the specific smoothing differences between the two approaches. Tenengrad values are provided into brackets for a quantitative comparison of sharpness.

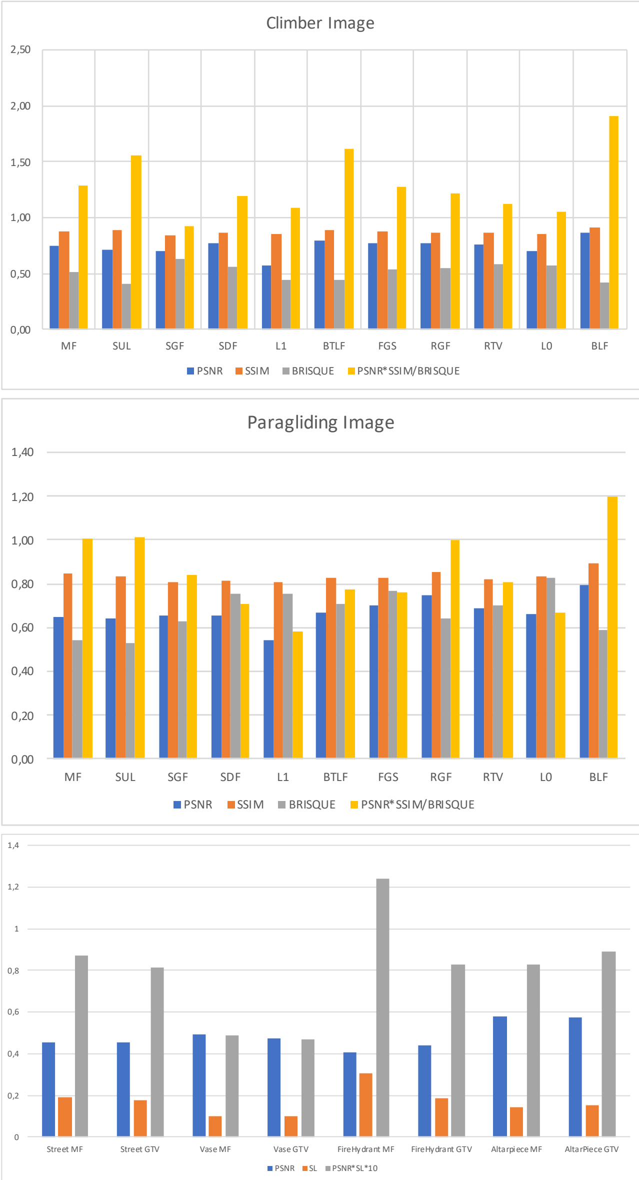


Fig. 9: Quantitative measures between methods for image and mesh smoothing. First line: results for Figure 6. Second line: results for Figure 7. Third line: results for Figure 8.

altogether. This is expressed by the following proposed scheme:

$$\hat{f}(v_k) = S_0(f_0(v_k)) + M(v_k) \cdot \sum_{i=1}^{l-1} S_i(f_i(v_k)). \qquad (10)$$

with $f_{l-1} = d_{l-1}$. Each layer is manipulated by a nonlinear s-curve function S_i that enables to perform both detail enhancement and tone manipulation. To avoid boosting noise and artefacts while enhancing the main structures of the original graph signal f, the layers are combined with the use of a structure mask M to enable a more focused detail manipulation. Let us now provide details on S_i and M.

4.2.2 Nonlinear boosting curve: Traditional approaches for detail manipulation manipulate the decomposition layers with specific linear coefficients [69, 70]:

$$S_i(x) = \alpha_i x \ . \qquad (11)$$

Unfortunately, this requires a very fine tuning of the parameters and can sometimes over-enhance some details. Some works have proposed to consider nonlinear detail manipulation coefficients [1, 71,

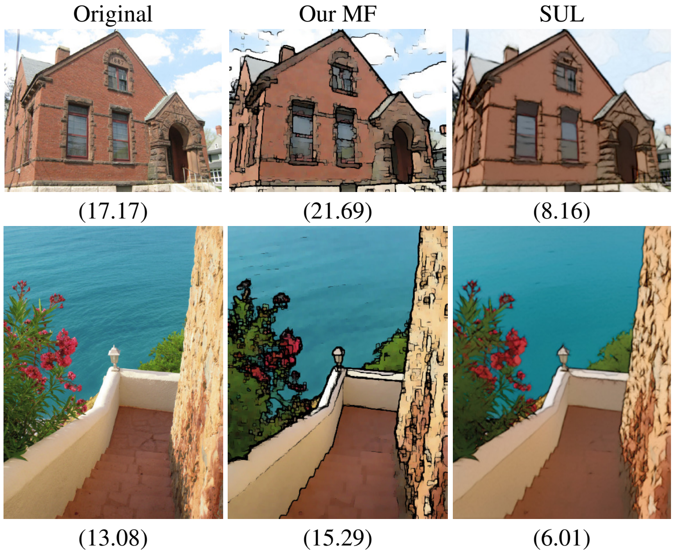


Fig. 10: Image abstraction result with our method (named MF) and SUL [61]. Tenengrad values are provided into brackets for a quantitative comparison of sharpness.

72]. In our proposed approach, a nonlinear function of sigmoid shape is considered:

$$S_i(x) = \frac{1}{1+exp(-\alpha_i x)} \ . \qquad (12)$$

The parameter α_i of the sigmoid is automatically determined and decreases while i increases. Given an initial parameter α, we set $\alpha_i = \frac{\alpha}{i+1}$.

4.2.3 Structure Mask: It has recently been put forward [72] that, for detail enhancement, it can be beneficial to selectively boost the most important image structures while keeping unmodified the rest of the image. For the case of graph signals, a vertex can be considered as an important structure if it is different from its neighbors. This amounts to say that the vertex has high spectral distances with its neighbors. In contrast, a vertex in an almost flat area will have low spectral distances with its neighbors. This can be used to construct a structure mask that differentiates the most salient structures in the graph signal. We propose to consider a normalized sum of the spectral distances between the vertex v_i and its direct neighbors $B_1(v_i)$ to construct this structure mask. It is defined as:

$$\nu(v_i) = \frac{\sum\limits_{v_j \in B_1(v_i)} d_{EMD}(H(v_j), H(v_i))}{|B_1(v_i)|} \qquad (13)$$

The considered distance d_{EMD} is the Earth Mover Distance [73]. It compares two histogram signatures that are compact representations of local distributions. The histograms $H(v_i)$ are of size N and are constructed on the index graph signal I. One has $H(v_i) = \{(w_k, m_k)\}_{k=1}^{N}$ within the set $B_1(v_i)$ where m_k is the index of the k-th element and w_k its appearance frequency. It is important to note that $N \leq |B_1(v_i)|$ since two signatures can have different sizes. Indeed, identical values can be found within the set $B_1(v_i)$ and the graph can be irregular. To compute the EMD, ground distances are computed in the CIELAB color space. Finally, the structure mask of a graph signal is formulated as

$$M(v_i) = 1 + \frac{\nu(v_i) - \wedge\nu}{\vee\nu - \wedge\nu} \ . \qquad (14)$$

where $\vee\nu$ and $\wedge\nu$ respectively denote the maximum and minimum values of $\nu(v_i)$. For a vertex v_i, $M(v_i) \in [1, 2]$ and is close to 1 for flat areas and to 2 for important structures (such as edges). In Figure 11 two structure masks examples on two colored graph signals are presented (for an image and a 3D mesh).

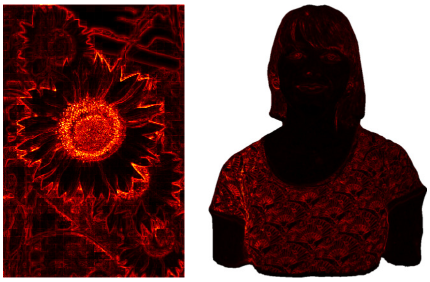


Fig. 11: Graph signal structure masks used to modulate the importance of detail enhancement (a heat color map is used to enhance the visualization). The original graph signals can be seen in other figures.

4.2.4 Detail manipulation examples: Now we provide some examples of graph signal detail manipulation. With our approach, we use 3 levels of multi-layer decomposition for detail manipulation. As for abstraction, in the state-of-the-art, the comparison is only visual. To ease the comparison of our approach with others, we consider the Tenengrad criterion [68, 69]. Figure 12 presents an example of detail manipulation with our approach without using structure mask, just only linear specific coefficients for each layer (Equation (11) is used). As can be seen, if our result has enhanced the original image, the enhancement is not as good as state-of-the-art approaches such as LLF [74]. This is mainly due to the fact that finding good coefficients for the detail manipulation is a very difficult task that depends on each image. However, when a structure mask is used with non-linear coefficients (Equation (12) is used), the result is fully automatic and with a standard tuning of the sole parameter of the method (α), a more visually pleasant result is obtained. As opposed to the results of LLF [74], no color aberrations and gradient reversal artefacts do appear in the background. Figure 13 provides complimentary comparison results. As can be seen, by varying the α parameter with $\alpha = 20$, we obtain results similar to BTF [7] and similar to LLF [74] with $\alpha = 30$ (but again without color aberrations and gradient reversal artefacts). Varying α results in tone manipulation. A strong advantage of our approach is that all the detail manipulation coefficients are automatically tuned whereas for the other approaches, they have been manually tuned to achieve the sharpest result. Finally, Figure 15 shows results for sharpness enhancement of 3D colored meshes and compares with the sole state-of-the-art method of Afrose *et al.* [60] that performs a processing similar to unsharp masking for images. For our morphological decomposition, the applied method is exactly the same as for images and this shows the versatility and generality of our approach that can process any type of graph signal. Figure 15 shows that the results of Afrose *et al.* [60] are often very subtle and difficult to see (even if the mesh looks globally sharper). In contrast, our results are sharper, as assessed by the Tenengrad values, and show a more pronounced detail manipulation. This is especially true for the vase and the person whereas for the two ancient building parts, results look very close.

4.3 Tone Mapping

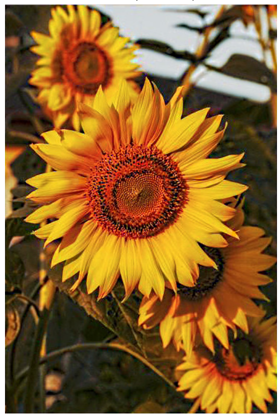
The third typical application of smoothing filters we consider is tone mapping. Indeed, as for detail manipulation, some state-of-the-art tone-mapping algorithms do need a prior decomposition into a base and a single detail layers. The approach we have proposed for multi-layer decomposition can also be considered to design a new morphological tone-mapping algorithm. To do so, we will replace the bilateral filter by our morphological one in the tone-mapping approach of Durand and Dorsey [75]. Let us recall the principle of this state-of-the-art approach. This method computes the log-luminance $L_l = \log(L(f))$ from an HDR image f with

$$L(f) = \frac{20 f_R + 40 f_G + f_B}{61} \tag{15}$$

Original (13.69)

LLF (25.09)

Our MF with linear coefficients (1, 1.25, 2.5) and without mask (20.19)



Our MF with mask ($\alpha = 30$) (24.21)

Fig. 12: Visual comparison between our method (named MF) and an image sharpening method, abbreviated as LLF [74]. Tenengrad values are provided into brackets for a quantitative comparison of sharpness.

where $f_{R,G,B}$ denotes a color channel of the HDR image. The log-luminance L_l image is then decomposed into a coarse base layer B_l and a detail layer obtained by the difference $D_l = L_l - B_l$. In the Durand and Dorsey approach [75], a bilateral filtering [10] is used to produce the base layer $B_l = \mathrm{BF}(L_l)$. The latter, as for any structure-preserving filter, is supposed to contain the main high contrast structures that need to be reduced. Therefore, this base layer is then compressed using gamma correction and finally added back to the detail layer. This is expressed by:

$$f'_* = \left(\frac{f_*}{L(f)} \exp\left(\tau B_l + (L_l - B_l)\right) \right)^{\frac{1}{\gamma}} \tag{16}$$

where f' is the RGB tone mapped image, τ is a contrast parameter and γ is a gamma correction parameter (fixed to 50 and 2.2 as recommended by Durand and Dorsey [75]). What we propose is to modify this tone mapping algorithm by replacing the bilateral filter with a morphological filter. However, we proceed differently due to our approach for morphological filtering that uses an index. Given an HDR image f, we use our decomposition to obtain a coarse base layer $f_0 = MF(f)$ and then compute its log-luminance. Therefore, one has $B_l = \log(L(MF(f)))$ instead (which is directly used in the approach of Durand and Dorsey [75]). In Figure 14, a comparison is shown between the state-of-the-art approach of Durand and Dorsey [75] and our morphological modification of the latter. It can be easily seen that in the base layer produced by the bilateral filter, many details have been preserved (see the stained glass) and edges have not always been well preserved (see the book and deck edges). In contrast, our base layer obtained by morphological decomposition is

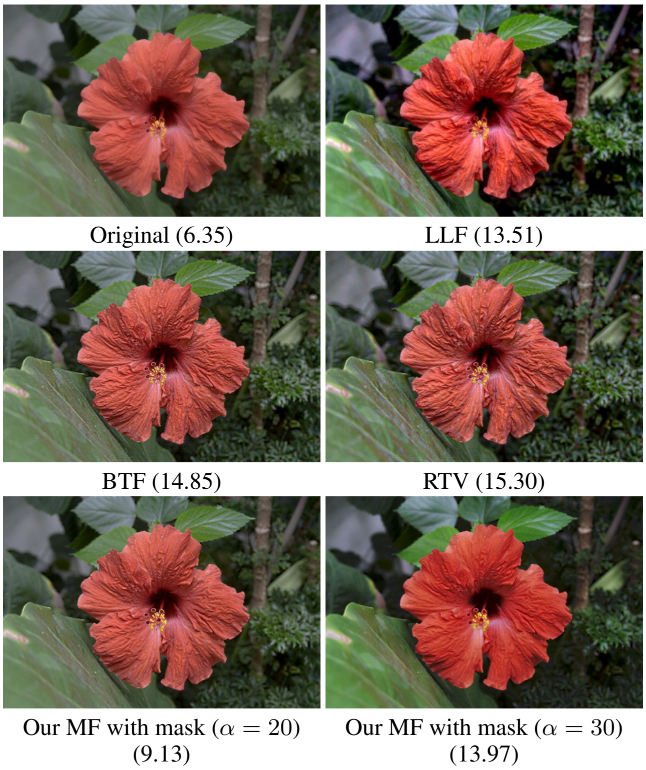


Fig. 13: Visual comparison between our method (named MF) and image sharpening methods, abbreviated as LLF [74], BTF [7] and RTV [64]. Tenengrad values are provided into brackets for a quantitative comparison of sharpness.

coarser and more piecewise constant. This is of high importance in the tone mapping algorithm and the final result we obtain is sharper and more contrasted than the one of Durand and Dorsey [75], as confirmed by the log-PSNR values (a quality metric for HDR images [76]*).

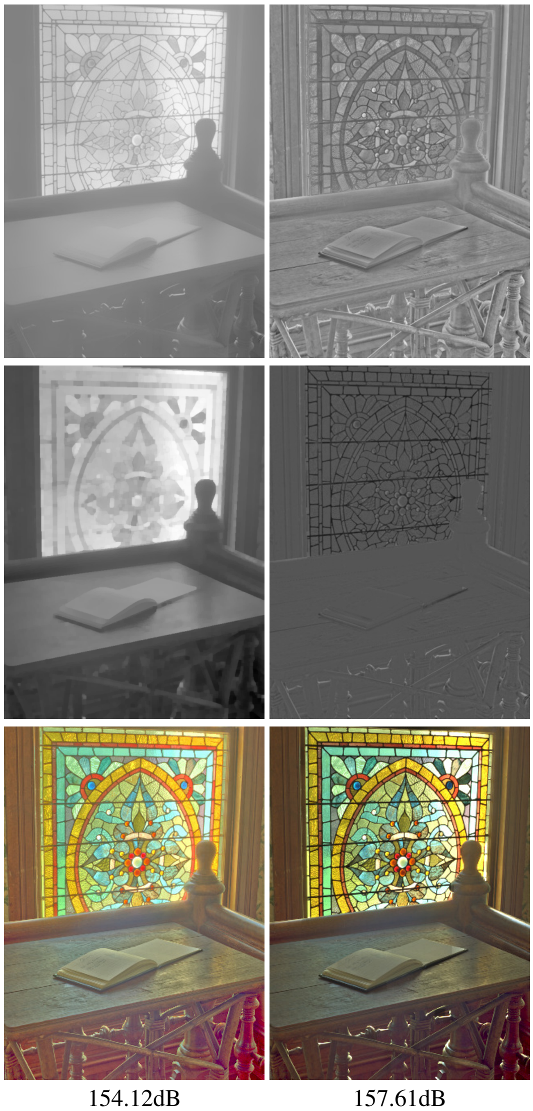


Fig. 14: Base and detail layers obtained with the approach of Durand and Dorsey [75] (first line) and our approach (second line). Last line shows the tone mapped image with the approach of Durand and Dorsey [75] (left) and our approach (right).

5 Conclusion

In this paper, we have considered the problem of computational editing of signal on graphs such as 2D color images and 3D colored meshes. Our approach considers morphological operators to perform the filtering. As morphological filters do require a complete lattice structure to be able to process vectorial data, a manifold-based complete lattice learning is devised. From this ordering, a new signal representation in the form of an index and a palette is obtained. Then, a hierarchical morphological multi-layer decomposition is proposed. We have shown how the latter can be used for editing applications such as abstraction, sharpness enhancement and tone mapping. The experimental results as well as their comparison with state-of-the-art algorithms have shown that our approach has some benefits and is competitive with existing approaches.

6 References

1 Farbman, Z., Fattal, R., Lischinski, D., Szeliski, R.: 'Edge-preserving decompositions for multi-scale tone and detail manipulation', *ACM Transactions on Graphics*, 2008, **27**, (3)

2 Aubert, G., Aujol, J.F.: 'Modeling very oscillating signals. application to image processing', *Applied Mathematics and Optimization*, 2005, **51**, (2), pp. 163–182

3 Gilles, J.: 'Multiscale texture separation', *Multiscale Modeling & Simulation*, 2012, **10**, (4), pp. 1409–1427

4 Hidane, M., Lézoray, O., Elmoataz, A.: 'Nonlinear multilayered representation of graph-signals', *Journal of Mathematical Imaging and Vision*, 2013, **45**, (2), pp. 114–137

5 Tadmor, E., Nezzar, S., Vese, L.: 'Multiscale hierarchical decomposition of images with applications to deblurring, denoising, and segmentation', *Commun Math Sci*, 2008, **6**, (2), pp. 281–307

6 Zhang, Q., Shen, X., Xu, L., Jia, J. 'Rolling guidance filter'. In: European Conference on Computer Vision. vol. 3, 2014. pp. 815–830

7 Cho, H., Lee, H., Kang, H., Lee, S.: 'Bilateral texture filtering', *ACM Transactions on Graphics*, 2014, **33**, (4), pp. 128:1–128:8

8 Gastal, E.S.L., Oliveira, M.M.: 'Domain transform for edge-aware image and video processing', *ACM Transactions on Graphics*, 2011, **30**, (4), pp. 69

9 Xu, L., Lu, C., Xu, Y., Jia, J.: 'Image smoothing via L_0 gradient minimization', *ACM Transactions on Graphics*, 2011, **30**, (6), pp. 174

10 Tomasi, C., Manduchi, R. 'Bilateral filtering for gray and color images'. In: ICCV, 1998. pp. 839–846

11 Min, D., Choi, S., Lu, J., Ham, B., Sohn, K., Do, M.N.: 'Fast global image smoothing based on weighted least squares', *IEEE Trans Image Processing*, 2014, **23**, (12), pp. 5638–5653

12 Bi, S., Han, X., Yu, Y.: 'An L_1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition', *ACM Trans Graph*, 2015, **34**, (4), pp. 78:1–78:12

13 Perona, P., Malik, J.: 'Scale-space and edge detection using anisotropic diffusion', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990, **12**, (7), pp. 629–639

14 Diop, E.H.S., Alexandre, R., Perrier, V.: 'A pde based and interpolation-free framework for modeling the sifting process in a continuous domain', *Advances in Computational Mathematics*, 2013, **38**, (4), pp. 801–835

15 Daubechies, I., Lu, J., Wu, H.T.: 'Synchrosqueezed wavelet transforms: An empirical mode decomposition-like tool', *Applied and Computational Harmonic Analysis*, 2011, **30**, (2), pp. 243 – 261

Available online at http://driiqm.mpi-inf.mpg.de

16 Diop, E.H.S., Alexandre, R., Boudraa, A..: 'Two-dimensional curvature-based analysis of intrinsic mode functions', *IEEE Signal Processing Letters*, 2018, **25**, (1), pp. 20–24
17 Diop, E.H.S., Alexandre, R., Moisan, L.: 'Intrinsic nonlinear multiscale image decomposition: A 2d empirical mode decomposition-like tool', *Computer Vision and Image Understanding*, 2012, **116**, (1), pp. 102 – 119
18 Colominas, M.A., Humeau-Heurtier, A., Schlotthauer, G.: 'Orientation-independent empirical mode decomposition for images based on unconstrained optimization', *IEEE Trans Image Processing*, 2016, **25**, (5), pp. 2288–2297
19 Wu, H.T.: 'Instantaneous frequency and wave shape functions (i)', *Applied and Computational Harmonic Analysis*, 2013, **35**, (2), pp. 181 – 199
20 Tavallali, P., Hou, T.Y., Shi, Z.: 'Extraction of intrawave signals using the sparse time-frequency representation method', *Multiscale Modeling & Simulation*, 2014, **12**, (4), pp. 1458–1493
21 Fleishman, S., Drori, I., Cohen-Or, D.: 'Bilateral mesh denoising', *ACM Transactions on Graphics*, 2003, **22**, (3), pp. 950–953
22 Subr, K., Soler, C., Durand, F.: 'Edge-preserving multiscale image decomposition based on local extrema', *ACM Trans Graph*, 2009, **28**, (5), pp. 147
23 Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Zheng, Q., et al.: 'The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis', *Proceedings of the Royal Society of London Series A: Mathematical, Physical and Engineering Sciences*, 1998, **454**, (1971), pp. 903–995
24 Serra, J., Vincent, L.: 'An overview of morphological filtering', *Circuits, Systems and Signal Processing*, 1992, **11**, (1), pp. 47–108
25 Cuisenaire, O.: 'Locally adaptable mathematical morphology using distance transformations', *Pattern Recognition*, 2006, **39**, (3), pp. 405 – 416
26 Lerallut, R., Decencière, E., Meyer, F.: 'Image filtering using morphological amoebas', *Image and Vision Computing*, 2007, **25**, (4), pp. 395 – 404
27 Verd-Monedero, R., Angulo, J., Serra, J.: 'Anisotropic morphological filters with spatially-variant structuring elements based on image-dependent gradient fields', *IEEE Transactions on Image Processing*, 2011, **20**, (1), pp. 200–212
28 Diop, E.H., Angulo, J.: 'Multiscale image analysis based on robust and adaptive morphological scale-spaces', *Image Analysis & Stereology*, 2014, **34**, (1), pp. 39–50
29 Meyer, F. 'Levelings and morphological segmentation'. In: Proceedings SIBGRAPI'98. International Symposium on Computer Graphics, Image Processing, and Vision (Cat. No.98EX237), 1998. pp. 28–35
30 Meyer, F., Maragos, P.: 'Nonlinear scale-space representation with morphological levelings', *Journal of Visual Communication and Image Representation*, 2000, **11**, (2), pp. 245 – 265
31 Angulo, J.: 'Levellings based on spatially adaptive scale spaces using local image features', *IET Image Processing*, 2019, **13**, pp. 1597–1607(10)
32 Brockett, R.W., Maragos, P.: 'Evolution equations for continuous-scale morphological filtering', *IEEE T Signal Proces*, 1994, **42**, (12), pp. 3377–3386
33 Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: 'Axioms and fundamental equations of image processing', *Archive for Rational Mechanics and Analysis*, 1993, **123**, (3), pp. 199–257
34 Serra, J.: 'Image Analysis and Mathematical Morphology'. vol. 2: Theoretical Advances. (Academic Press, 1988)
35 Peters, R.A.: 'A new algorithm for image noise reduction using mathematical morphology', *IEEE Transactions on Image Processing*, 1995, **4**, (5), pp. 554–568
36 Ronse, C.: 'Why mathematical morphology needs complete lattices', *Signal Processing*, 1990, **21**, (2), pp. 129–154
37 Chevallier, E., Angulo, J.: 'The irregularity issue of total orders on metric spaces and its consequences for mathematical morphology', *J MATH IMAGING VIS*, 2016, **54**, (3), pp. 344–357
38 Lézoray, O.: 'Complete lattice learning for multivariate mathematical morphology', *Journal of Visual Communication and Image Representation*, 2016, **35**, pp. 220–235
39 Shuman, D.I., Narang, S.K., Frossard, P., Ortega, A., Vandergheynst, P.: 'The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains', *IEEE Signal Process Mag*, 2013, **30**, (3), pp. 83–98
40 Barnett, V.: 'The ordering of multivariate data', *Journal of the Royal Statistical Society Series A (General)*, 1976, **139**, (3), pp. 318–355
41 Aptoula, E., Lefèvre, S.: 'A comparative study on multivariate mathematical morphology', *Pattern Recognit*, 2007, **40**, (11), pp. 2914–2929
42 Aptoula, E., Lefèvre, S.: 'On lexicographical ordering in multivariate mathematical morphology', *Pattern Recognition Letters*, 2008, **29**, (2), pp. 109–118
43 Burgeth, B., Kleefeld, A. 'Morphology for color images via loewner order for matrix fields'. In: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing, 2013. pp. 243–254
44 Breuß, M., Hoeltgen, L., Kleefeld, A. 'Matrix-valued levelings for colour images'. In: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing, 2017. pp. 296–308
45 Goutsias, J., Heijmans, H.J.A.M., Sivakumar, K.: 'Morphological operators for image sequences', *Computer Vision and Image Understanding*, 1995, **62**, (3), pp. 326–346
46 Velasco.Forero, S., Angulo, J.: 'Random projection depth for multivariate mathematical morphology.', *IEEE Journal of Selected Topics in Signal Processing*, 2012, **6**, (7), pp. 753–763
47 Velasco.Forero, S., Angulo, J. 'Vector ordering and multispectral morphological image processing'. In: Celebi, M.E., Smolka, B., editors. Advances in Low-Level Color Image Processing. vol. 11 of *Lecture Notes in Computational Vision and Biomechanics*. (Springer Netherlands, 2014. pp. 223–239
48 Aptoula, E., Lefèvre, S. 'Multivariate mathematical morphology applied to colour image analysis'. In: Collet, C., Chanussot, J., Chehdi, K., editors. Multivariate image processing: methods and applications. (ISTE - John Wiley, 2009. pp. 303–337
49 Chanussot, J., Lambert, P. 'Bit mixing paradigm for multivalued morphological filters'. In: International Conference on Image Processing and Its Applications. vol. 2, 1997. pp. 804 – 808
50 Lezoray, O., Charrier, C., Elmoataz, A. 'Rank transformation and manifold learning for multivariate mathematical morphology'. In: EUSIPCO (European Signal Processing Conference), 2009. pp. 35–39
51 Peyré, G.: 'Manifold models for signals and images', *Computer Vision and Image Understanding*, 2009, **113**, (2), pp. 249–260
52 Chanussot, J., Lambert, P. 'Total ordering based on space filling curves for multivalued morphology'. In: Proceedings of the Fourth International Symposium on Mathematical Morphology and Its Applications to Image and Signal Processing, 1998. pp. 51–58
53 Lee, J.A., Verleysen, M.: 'Nonlinear Dimensionality Reduction'. (Springer, 2007)
54 Talwalkar, A., Kumar, S., Mohri, M., Rowley, H.A.: 'Large-scale SVD and manifold learning', *Journal of Machine Learning Research*, 2013, **14**, (1), pp. 3129–3152
55 Gersho, A., Gray, R.M.: 'Vector Quantization and Signal Compression'. (Kluwer Academic, 1991)
56 Heijmans, H.J.A.M., Nacken, P.F.M., Toet, A., Vincent, L.: 'Graph morphology', *J Visual Communication and Image Representation*, 1992, **3**, (1), pp. 24–38
57 Hidane, M., Lézoray, O., Elmoataz, A. 'Graph signal decomposition for multi-scale detail manipulation'. In: International Conference on Image Processing (IEEE), 2014. pp. 2041–2045
58 Lézoray, O., Grady, L.: 'Image Processing and Analysis with Graphs: Theory and Practice'. Lézoray, O., Grady, L., editors. Digital Imaging and Computer Vision. (CRC Press / Taylor and Francis, 2012)
59 Lézoray, O. '3d colored mesh graph signals multi-layer morphological enhancement'. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017. pp. 1358–1362
60 Afrose, Z., Shen, Y.: 'Mesh color sharpening', *Advances in Engineering Software*, 2016, **91**, pp. 36 – 43
61 Fan, Q., Yang, J., Wipf, D.P., Chen, B., Tong, X.: 'Image smoothing via unsupervised learning', *ACM Trans Graph*, 2018, **37**, (6), pp. 259:1–259:14
62 Zhang, F., Dai, L., Xiang, S., Zhang, X. 'Segment graph based image filtering: Fast structure-preserving smoothing'. In: IEEE International Conference on Computer Vision, 2015. pp. 361–369
63 Ham, B., Cho, M., Ponce, J. 'Robust image filtering using joint static and dynamic guidance'. In: IEEE Conference on Computer Vision and Pattern Recognition, 2015. pp. 4823–4831
64 Xu, L., Yan, Q., Xia, Y., Jia, J.: 'Structure extraction from texture via relative total variation', *ACM Trans Graph*, 2012, **31**, (6), pp. 139:1–139:10
65 Wang, Z., Bovik, A.C.: 'Mean squared error: Love it or leave it? a new look at signal fidelity measures', *IEEE Signal Processing Magazine*, 2009, **26**, (1), pp. 98–117
66 Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: 'Image quality assessment: from error visibility to structural similarity', *IEEE Trans Image Processing*, 2004, **13**, (4), pp. 600–612
67 Mittal, A., Moorthy, A.K., Bovik, A.C.: 'No-reference image quality assessment in the spatial domain', *IEEE Trans Image Processing*, 2012, **21**, (12), pp. 4695–4708
68 Xu, X., Wang, Y., Zhang, X., Li, S., Liu, X., Wang, X., et al.: 'A comparison of contrast measurements in passive autofocus systems for low contrast images', *Multimedia Tools and Applications*, 2014, **69**, (1), pp. 139–156
69 Choudhury, A., Medioni, G. 'Perceptually motivated automatic sharpness enhancement using hierarchy of non-local means'. In: IEEE International Conference on Computer Vision Workshops, ICCV, 2011. pp. 730–737
70 Choudhury, A., Medioni, G.: 'Hierarchy of nonlocal means for preferred automatic sharpness enhancement and tone mapping', *Journal of the Optical Society of America A*, 2013, **30**, (3), pp. 353–366
71 Paris, S., Hasinoff, S.W., Kautz, J.: 'Local laplacian filters: edge-aware image processing with a laplacian pyramid', *ACM Transactions on Graphics*, 2011, **30**, (4), pp. 68
72 Esfandarani, H.T., Milanfar, P.: 'Fast multilayer laplacian enhancement', *IEEE Trans Computational Imaging*, 2016, **2**, (4), pp. 496–509
73 Rubner, Y., Tomasi, C., Guibas, L.J.: 'The earth mover's distance as a metric for image retrieval', *International Journal of Computer Vision*, 2000, **40**, (2), pp. 99–121
74 Paris, S., Hasinoff, S.W., Kautz, J.: 'Local laplacian filters: edge-aware image processing with a laplacian pyramid', *Commun ACM*, 2015, **58**, (3), pp. 81–91
75 Durand, F., Dorsey, J.: 'Fast bilateral filtering for the display of high-dynamic-range images', *ACM Trans Graph*, 2002, **21**, (3), pp. 257–266
76 Mantiuk, R.K., Myszkowski, K., Sidel., H.P. 'High dynamic range imaging'. In: Wiley Encyclopedia of Electrical and Electronics Engineering. (Wiley, 2015. pp. 1–42

Acknowledgments

This work received funding from the Agence Nationale de la Recherche (ANR-14-CE27-0001 GRAPHSIP), from the European Union FEDER/FSE 2014/2020 (GRAPHSIP project), and the support of NVIDIA Corporation with the donation of a Titan Xp GPU used for this research. We thank the reviewers for their constructive comments that have much enabled to improve the paper.

Fig. 15: Visual comparison between our method (named MF) for 3D colored mesh detail manipulation and unsharp masking [60]. Tenengrad values are provided into brackets for a quantitative comparison of sharpness.