Patch-Based Mathematical Morphology for Image Processing, Segmentation and Classification

Olivier Lezoray

▶ **To cite this version:**

Olivier Lezoray. Patch-Based Mathematical Morphology for Image Processing, Segmentation and Classification. Advanced Concepts for Intelligent Vision Systems, 2015, Catane, Italy. pp.46-57, 10.1007/978-3-319-25903-1_5 . hal-01252874

HAL Id: hal-01252874
https://hal.science/hal-01252874

Submitted on 11 Jan 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Patch-based Mathematical Morphology for image processing, segmentation and classification

Olivier Lézoray

Normandie Univ., UNICAEN, ENSICAEN, GREYC UMR CNRS 6072, Caen, France
olivier.lezoray@unicaen.fr

Abstract. In this paper, a new formulation of patch-based adaptive mathematical morphology is addressed. In contrast to classical approaches, the shape of structuring elements is not modified but adaptivity is directly integrated into the definition of a patch-based complete lattice. The manifold of patches is learned with a nonlinear bijective mapping, interpreted in the form of a learned rank transformation together with an ordering of vectors. This ordering of patches relies on three steps: dictionary learning, manifold learning and out of sample extension. The performance of the approach is illustrated with innovative examples of patch-based image processing, segmentation and texture classification.

1 Introduction

Mathematical Morphology (MM) is a powerful framework for nonlinear processing of images. Morphological operators are usually defined by using the concept of Structuring Elements (SEs), small subsets used to explore images. The output of a morphological filtering operation is then obtained by the interaction between the image and a given SE. This idea has been extended to grey scale images using the concept of complete lattices (orderings between the elements to be processed) and MM relies on the application of lattice theory to spatial structures in images. In classical MM, SEs remain the same for all points in the image domain, i.e., one single SE is used to process the whole image by translating it to every point in the image. Adaptive MM refers to morphological filtering techniques that adjust SEs to the local context of the image. With the need of more efficient morphological image processing operators, there has been recently much interest in the development of adaptive mathematical morphology (see [8] for a recent survey). Roughly, two types of adaptive MM can be considered [17, 8, 14]: i) location-adaptive MM: the shape of the structuring element depends on the location x in the image, ii) input-adaptive MM: the shape of the structuring element depends on local features extracted at the location x. In the same time, image processing using local patches has become very popular and was shown to be highly effective [6]. The processing proceeds by operating on the image patches and exploiting their similarities, making the processing much more adaptive to the image. Patches being simply feature vectors locally describing a pixel at a given location in the image, some authors have considered the use of patches within adaptive morphological operators.

Ta *et al.* [20–22] were the first to propose the use of patches for MM processing. They have proposed a framework for adapting continuous MM on discrete graph structures and the adaptivity is input-adaptive at two levels: the shape of the SE is expressed by the graph topology that depends on a patch nearest neighbor graph, and the PDE morphological process is also adaptive by incorporating weights into it. Other works have followed and considered the algebraic formulation of MM. In [19], Salembier has introduced flat MM with adaptive SE obtained from a patch nearest neighbor graph (this is a special case of [20–22]) and non-flat MM with patch similarities incorporated into the SE. Then, Velasco-Forero and Angulo [25] have recasted the works of Salembier in the general framework of adaptive MM and presented their necessary properties to be considered as algebraic MM operators. Recently, in [26] Yang and Li have considered a new type of adaptive SE based on amoeba SE combining local geodesic distance and non-local patch distance for spatially variant morphological filters.

In this paper we consider a radically different approach for patch-based adaptive MM. Indeed, in these patch-adaptive MM approaches the shape of the SE is classically modified to account for patch similarities. Since MM is based on complete lattices, we can instead integrate the adaptivity directly into the definition of the complete lattice and therefore define an ordering relationship between patches. To deal with this difficult objective, we build upon our previous work [13] that constructs complete lattices in vector spaces as a rank transform learned through a nonlinear mapping. In the next section we show how complete lattice construction by learning the patch manifold can be performed. Then we show the benefit of our approach for patch-based morphological image processing, segmentation and texture classification.

2 Patch Complete Lattice Learning

2.1 Complete lattice from patches

An image is represented by the mapping $f : \Omega \subset \mathbb{Z}^l \to \mathcal{T} \subset \mathbb{R}^n$ where l is the image dimension, n the number of channels, and $\mathcal{T}$ is a non-empty set of the image multivariate vectors. To each pixel $x_i \in \Omega$ of an image is associated a vector $\mathbf{v}_i = f(x_i)$. We denote as $\mathcal{P}$ the vector space of patches of width w associated to pixels of f, which is represented as the mapping: $F_w : \Omega \subset \mathbb{Z}^l \to \mathcal{P} \subset \mathbb{R}^{nw^2}$. One has $\mathbf{p}_i^w = F_w(x_i) = \big(f(x_i + t), \forall t \in [-w/2, w/2]^2\big)^T$. Performing MM operations for functions on patch vector spaces therefore requires the definition of a complete lattice $(\mathcal{P}, \leq)$ [18] which means that we have to be able to compare patches to order them. Comparing color vectors to define color complete lattice being already difficult [3], one easily see that defining a complete lattice for patches' vectors is much more challenging (a classical lexicographic ordering [2] being obviously of no interest). One way to define an ordering relation between vectors of a set $\mathcal{T}$ is to use the framework of h-orderings [10]. This corresponds to defining a surjective transform h from $\mathcal{T}$ to $\mathcal{L}$ where $\mathcal{L}$ is a complete lattice equipped with the conditional total ordering [10]. We refer to $\leq_h$ as the h-ordering given

by:

$$h : \mathcal{T} \rightarrow \mathcal{L} \text{ and } \mathbf{v} \rightarrow h(\mathbf{v}), \forall (\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{T} \times \mathcal{T}, \quad \mathbf{v}_i \leq_h \mathbf{v}_j \Leftrightarrow h(\mathbf{v}_i) \leq h(\mathbf{v}_j) \ . \quad (1)$$

Then, $\mathcal{T}$ is no longer required to be a complete lattice, since the ordering of $\mathcal{T}$ can be induced upon $\mathcal{L}$ by means of h [3]. When h is bijective, this corresponds to defining a space filling curve [7] or equivalently a rank transform [12]. We propose to adapt the h-ordering framework to our problem of complete lattice construction from patches for morphological image processing. Since a unique patch $\mathbf{p}_i^w \in \mathcal{P}$ is associated to a given vector $\mathbf{v}_i \in \mathcal{T}$ (and vice-versa), a complete lattice $(\mathcal{T}, \leq_h^w)$ can be directly deduced from a complete lattice of patches $(\mathcal{P}, \leq_h^w)$: $\mathbf{v}_i \leq_h^w \mathbf{v}_j \Leftrightarrow \mathbf{p}_i^w \leq_h^w \mathbf{p}_j^w$. This means that vectors of $\mathcal{T}$ can be ordered using a patch comparison and we obtain a patch adaptive complete lattice definition for all the vectors of the image. We denote by $\mathbf{p}^w(\mathbf{v}_i)$ the patch of width w associated to a vector $\mathbf{v}_i$. We obtain the following patch-based complete lattice $(\mathcal{T}, \leq_h^w)$ definition:

$$h^w : \mathcal{T} \rightarrow \mathcal{L} \text{ and } \mathbf{v} \rightarrow h(\mathbf{p}^w(\mathbf{v}))$$

$$\forall (\mathbf{p}^w(\mathbf{v}_i), \mathbf{p}^w(\mathbf{v}_j)) \in \mathcal{P} \times \mathcal{P} \text{ associated to } (\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{T} \times \mathcal{T}$$

$$\mathbf{v}_i \leq_h \mathbf{v}_j \Leftrightarrow h^w(\mathbf{p}^w(\mathbf{v}_i)) \leq h^w(\mathbf{p}^w(\mathbf{v}_j)) \ . \quad (2)$$

The question is now on how to construct the mapping h^w to compare patches. It is obvious that h^w cannot be linear [16] since a distortion of the space topology is inevitable. As a consequence, we choose to focus our developments on learning the patch manifold to construct h^w to compare patches.

2.2 Complete Lattice learning

We show how to construct a h^w-ordering for patches extracted from an image. This is an adaptation of our previous works [13] and we summarize its principle in the sequel, the whole approach being detailed in the form of an algorithm in Algorithm 1. The approach consists in learning the manifold of patches with a non linear mapping from a given image and to define the patch h^w-ordering from this projection. To learn the manifold of patches, we use Laplacian EigenMaps (LE), a technique for non-linear dimensionality reduction [11]. Computationally, performing LE on the whole space of patches is not tractable in reasonable time, so we use a four-step strategy that enables us to construct efficiently a h^w-ordering. Given an image $f : \Omega \rightarrow \mathcal{T} \subset \mathbb{R}^n$ that provides a set $\mathcal{T}$ of m vectors in $\mathbb{R}^n$, a sampling (both regular and random) is performed on the set $\mathcal{P}$ of all m patches of f to obtain a smaller $\mathcal{P}'$ (but representative) set of m' patches. From $\mathcal{P}'$, a dictionary $\mathcal{D} = \{\mathbf{x}'_1, \cdots, \mathbf{x}'_p\}$ of p vectors is build by Vector Quantization [9]. Manifold learning by Laplacian EigenMaps is performed on this dictionary. One starts by computing a similarity matrix $\mathbf{K}_\mathcal{D}$ that contains the pairwise similarities $K_\mathcal{D}(i,j)$ between all the dictionary vectors $\mathbf{x}'_i$. To have a parameter-free algorithm, σ is set to the maximum distance between input vectors. The normalized Laplacian matrix $\mathbf{L} = \mathbf{I} - \mathbf{D}_\mathcal{D}^{-\frac{1}{2}} \mathbf{K}_\mathcal{D} \mathbf{D}_\mathcal{D}^{-\frac{1}{2}}$ is then computed. Laplacian

Eigenmaps Manifold Learning consists in searching for a new representation $\mathbf{Y}$ obtained by minimizing $\frac{1}{2}\sum_{ij}\left\|\mathbf{y}_i-\mathbf{y}_j\right\|_2 K_{\mathcal{D}}(i,j) = Tr(\mathbf{Y}^T\mathbf{L}\mathbf{Y})$ under the constraint $\mathbf{Y}^T\mathbf{D}\mathbf{Y}=\mathbf{I}$. This cost function encourages nearby sample vectors to be mapped to nearby outputs. The solution is obtained [5] by finding the eigenvectors $\mathbf{\Phi}_{\mathcal{D}}$ of $\mathbf{L}$. This obtained projection operator corresponds to constructing a $h_{\mathcal{D}}$-ordering from the data of the dictionary $\mathcal{D}$ and a new representation $h^w_{\mathcal{D}}(\mathbf{x}'_i)$ is obtained for each element $\mathbf{x}'_i$ of the dictionary:

$$h^w_{\mathcal{D}} : \mathbf{x}'_i \rightarrow (\boldsymbol{\phi}^1_{\mathcal{D}}(\mathbf{x}'_i), \cdots, \boldsymbol{\phi}^p_{\mathcal{D}}(\mathbf{x}'_i))^T \in \mathbb{R}^p \ . \tag{3}$$

Such a strategy of modeling the manifold from a patch dictionary was also explored in [16]. This correspond to the construction of the complete lattice $(\mathcal{D}, \leq_{h^w_{\mathcal{D}}})$ with a $h^w_{\mathcal{D}}$-ordering, and this ordering is only valid for the set of patches of the dictionary. Since we need the complete lattice $(\mathcal{P}, \leq^w_h)$, the reduced dictionary lattice is extended to all the patches of the initial lattice $\mathcal{P}$ by Nyström extrapolation [23] of $h^w_{\mathcal{D}}$ on $\mathcal{P}$, and the complete lattice $(\mathcal{P}, \leq^w_h)$ is obtained as $h^w : \mathbf{p}^w(\mathbf{v}_i) \rightarrow (\boldsymbol{\phi}^1(\mathbf{p}^w(\mathbf{v}_i)), \cdots, \boldsymbol{\phi}^p(\mathbf{p}^w(\mathbf{v}_i)))^T \in \mathbb{R}^p$. From this complete lattice on patches, is deduced the patch-based complete lattice $(\mathcal{T}, \leq^w_h)$ on the initial vectors of f.

Algorithm 1 Learning the Patch-based h^w-ordering

Inputs:

Image $f : \Omega \subset \mathbb{Z}^l \rightarrow \mathcal{T} \subset \mathbb{R}^n$

Set $\mathcal{T}$ of m input multivariate vectors $\mathbf{v}_i$ of f

Set $\mathcal{P}$ of m patches $\mathbf{p}^w(\mathbf{v}_i)$ extracted from f

Step 1: Patch Sampling

Construct from $\mathcal{P}$, by sampling, a new set $\mathcal{P}' = \{\mathbf{x}_1, \ldots, \mathbf{x}_{m'}\} \subset \mathcal{P}$ of m' patches

Step 2: Patch Dictionary Construction

Build from $\mathcal{P}'$, by VQ, a patch dictionary $\mathcal{D} = \{\mathbf{x}'_1, \ldots, \mathbf{x}'_p\}$ with $p \ll m'$

Step 3: Patch Manifold Learning on the dictionary

Compute the similarity matrix $\mathbf{K}_{\mathcal{D}}$ between vectors $\mathbf{x}'_i \in \mathcal{D}$ with

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

Compute the degree diagonal matrix $\mathbf{D}_{\mathcal{D}}$ of $\mathbf{K}_{\mathcal{D}}$

Compute the eigen-decomposition of the normalized Laplacian

$\mathbf{L} = \mathbf{I} - \mathbf{D}_{\mathcal{D}}^{-\frac{1}{2}}\mathbf{K}_{\mathcal{D}}\mathbf{D}_{\mathcal{D}}^{-\frac{1}{2}}$ as $\mathbf{L} = \mathbf{\Phi}_{\mathcal{D}}\mathbf{\Pi}_{\mathcal{D}}\mathbf{\Phi}_{\mathcal{D}}^T$

with eigenvectors $\mathbf{\Phi}_{\mathcal{D}} = [\mathbf{\Phi}^1_{\mathcal{D}}, \cdots, \mathbf{\Phi}^p_{\mathcal{D}}]$ and eigenvalues $\mathbf{\Pi}_{\mathcal{D}} = \mathrm{diag}[\lambda_1, \cdots, \lambda_p]$

Step 4: Extrapolation of the projection $\mathbf{\Phi}_{\mathcal{D}}$ to all the patches of $\mathcal{P}$

Compute similarity matrices $\mathbf{K}_{\mathcal{P}}$ on $\mathcal{P}$ and $\mathbf{K}_{\mathcal{D}\mathcal{P}}$ between sets $\mathcal{D}$ and $\mathcal{P}$

Compute the degree diagonal matrix $\mathbf{D}_{\mathcal{D}\mathcal{P}}$ of $\mathbf{K}_{\mathcal{D}\mathcal{P}}$

Extrapolate eigenvectors obtained from $\mathcal{D}$ to $\mathcal{P}$ with

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

Output:

The projection $h_w : \mathcal{T} \subset \mathbb{R}^n \rightarrow \mathcal{L} \subset \mathbb{R}^p$ is given by $\mathbf{\Phi}$ and defines the h^w-ordering.

2.3 Patch-based MM operators

Given the patch-based complete lattice $(\mathcal{T}, \leq_h^w)$, we sort all vectors of f according to $\leq_h^w$ (the conditional total ordering on $h^w(\mathbf{x})$) and obtain a sorted image f_h^w. This sorted image $f_h^w : [1, m] \to \mathbb{R}^n$ defines the ordering of the vectors of f. This corresponds to a view of the learned complete lattice $(\mathcal{T}, \leq_h^w)$. From this ordering, we can deduce the rank of a vector on the complete lattice $\mathcal{L}$ defined as $r : \mathbb{R}^p \to [1, m]$, and construct a rank image as

$$f_r : \Omega \to [1, m], \text{ with } f_r(x_i) = (r \circ h^w \circ f)(x_i), \forall x_i \in \Omega \ . \tag{4}$$

In addition, we have also the definition of the inverse

$$(h^w)^{-1}(x_i) = (f_h^w \circ r)(x_i), \forall x_i \in \Omega \tag{5}$$

which is unique. With these elements, the original image f is now represented by the rank image f_r and the ordering of the pixels' vectors f_h^w. The original image f is recovered exactly since $f(x_i) = (f_h^w \circ f_r)(x_i), \forall x_i \in \Omega$. This shows that each pixel x_i vector is recovered by getting its corresponding vector in the Look-Up-Table f_h^w with the index $f_r(x_i)$. Given a specific morphological processing g, the corresponding processed multivariate image is obtained by

$$g(f(x_i)) = (f_h^w \circ g \circ f_r)(x_i), \forall x_i \in \Omega \ . \tag{6}$$

We can now formulate the corresponding h^w-erosion $\epsilon_{h^w,B}$ and h^w-dilation $\delta_{h^w,B}$ of an image f at pixel $x_i \in \Omega$ by the structuring element $B \subset \Omega$ as:

$$\epsilon_{h^w,B}(f)(x_i) = \{f_h^w(\wedge f_r(p_j)), p_j \in B(x)\} = \{f_h^w(\epsilon_B(f_r)(x_i))\} \tag{7}$$

and

$$\delta_{h^w,B}(f)(x_i) = \{f_h^w(\vee f_r(p_j)), p_j \in B(x)\} = \{f_h^w(\delta_B(f_r)(x_i))\} \tag{8}$$

with ϵ_B and δ_B the classical erosion and dilation on scalar images. This shows that the MM operators operate on the ranks f_r, and the image is reconstructed through the sorted vectors f_h^w that represent the learned lattice. It is easy to see that these operators inherit the standard algebraic properties of morphological operators since they fit into the theory of h-adjunctions [24]. From these basic operators, we can obtain many morphological filters such as the h^w-openings and h^w-closings:

$$\gamma_{h^w,B}(f) = \delta_{h^w,B}(\epsilon_{h^w,B}(f)) = f_h^w(\delta_B(\epsilon_B(f_r))) \tag{9}$$

$$\phi_{h^w,B}(f) = \epsilon_{h^w,B}(\delta_{h^w,B}(f)) = f_h^w(\epsilon_B(\delta_B(f_r)) \tag{10}$$

3 Applications

To illustrate the benefit of the approach, we provide several examples of its use for morphological image processing, image segmentation and texture classification. In all the experiments, the number of elements of the dictionary $\mathcal{D}$ depends on the number m' of vectors sampled from the original image and it is automatically fixed to $p = 2^k$ with k the largest integer value such that $2^k \leq \sqrt{m'}/8$. When colors are considered instead of patches (in this case $w = 1$), no sampling is performed and $m' = m$.

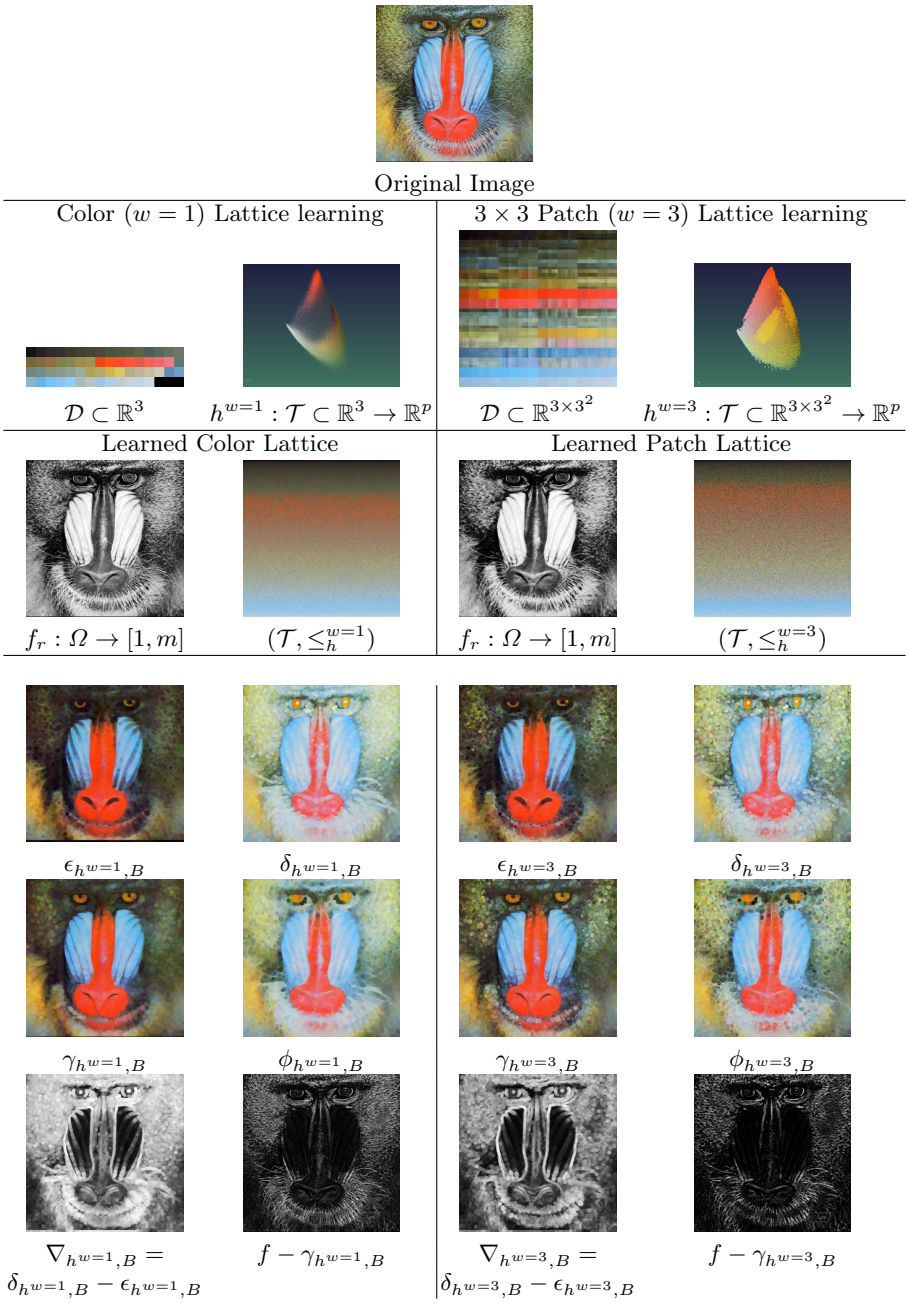

Fig. 1. Morphological processing of color images with a learned complete lattice from colors ($h^{w=1}$) or 3×3 patches ($h^{w=3}$). The structuring element is a circle of radius 5.

3.1 Color image processing

To illustrate our approach, we consider a color image $f : \Omega \to \mathcal{T} \subset \mathbb{R}^3$. The complete lattice is learned from the image and we obtain both rank $f_r : \Omega \to [1, m]$ and ordering $f_h^w : [1, m] \to \mathbb{R}^3$. Then, we compute the following morphological operators: h^w-erosion $\epsilon_{h^w,B}$, h^w-dilation $\delta_{h^w,B}$, h^w-opening $\gamma_{h^w,B}$, h^w-closing $\phi_{h^w,B}(f)$, h^w-morphological gradient $\nabla_{h^w,B}(f) = \delta_B(f_r) - \epsilon_B(f_r)$, and h^w-white top hat. To see the effect of using a patch lattice instead of of color lattice, we learn the lattice either directly from color vectors ($w = 1$) or color 3×3 patches ($w = 3$). Figure 1 presents the results. Second line shows the dictionary $\mathcal{D}$ and the extrapolated manifold eigenvectors $\mathbf{\Phi}$ (shown on the three first axis). Third line presents the induced learned lattice illustrated by the rank and the ordering. As it can be seen, with the learned color lattice, we recover the classical aspects of MM operators: erosion contracts structures of color far from first color (black) of the complete lattice. Dilation provides the dual effect and extends structures of color close to last color (white) of the complete lattice. If we now compare the results between a color and a patch lattice, with a patch-based ordering, the simplification effect is less strong and texture is much better preserved and sharper results are obtained. Meanwhile the patch-based morphological processing still exhibits the dual effect between both opening and closing filters. Finally, patch-based gradient and white top hat provide much contrasted results than color ones. Figure 2 presents results of an opening by reconstruction with the classical lexicographic ordering and our proposed learned complete lattice from color and patches. The patch-based processing shows again much better results: the images have been strongly simplified but the color and texture are much coherent and better preserved than with the other lattices.

Fig. 2. Opening by reconstruction of color images by lexicographic color lattice (h^{Lex}) or learned complete lattice from color ($h^{w=1}$) and 5×5 patches ($h^{w=5}$). The marker image is obtained from an erosion in the same lattice with a square structuring element of side 11 pixels.

3.2 Color image segmentation

To further show the interest of a patch-based processing, we consider its application for image segmentation. Figure 3 presents such results. From the original image, morphological gradients are computed with the classical lexicographic ordering and our proposed approach from color and patches of different sizes. One can see (first row of Figure 3) on the patch based gradient images that, in areas of similar textures no high gradient values are found whereas in the color based gradients, high gradient values are found at strong color variations. In addition, the patch-based gradients become smoother as the patch size grows, assessing the capture of larger texture cues. Then, region seeds are superimposed interactively (second row of Figure 3). Using the gradients and the seeds, a marker controlled watershed is computed on the considered gradients (last row of Figure 3). The interest of a patch based processing appears then evident since it enables to obtain a smoother and more precise segmentation.

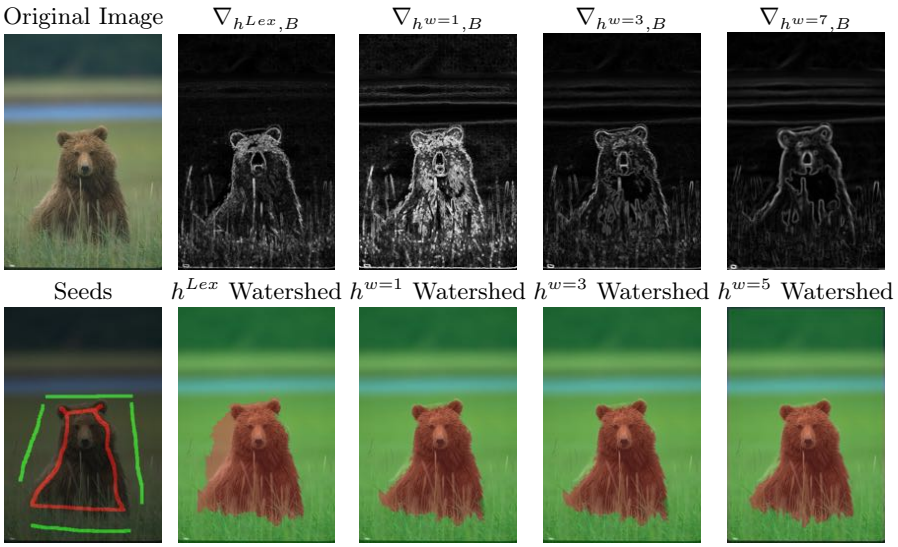

Fig. 3. Segmentation of a color image with a seeded watershed on a morphological gradient computed from lexicographic color lattice (h^{Lex}) or learned complete lattice from color ($h^{w=1}$), 3×3 patches ($h^{w=3}$), and 7×7 patches ($h^{w=7}$).

Figure 4 presents three additional MM segmentation from seeds. Only large patches are now considered as they exhibited good results in Figure 3. The results of the learned color lattice are slightly better than with lexicographic ordering, but a patch based lattice enables to better delineate the objects' contours, even in very difficult images such as the tiger one.

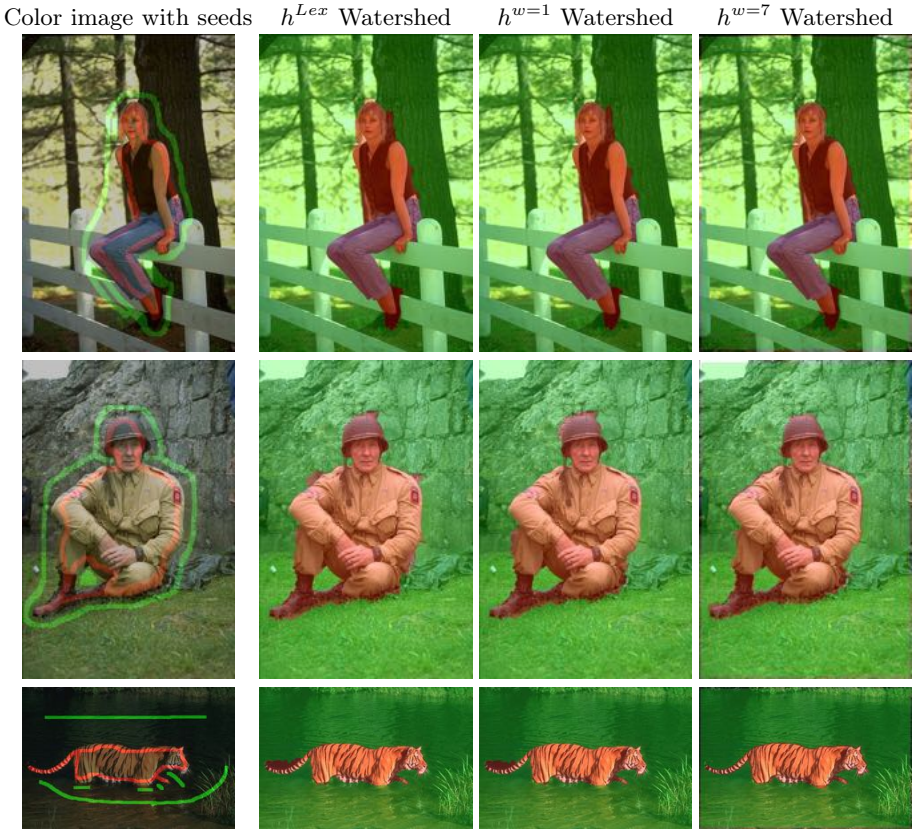

Fig. 4. Color image segmentation with a seed watershed from lexicographic color lattice (h^{Lex}) or learned complete lattice from color ($h^{w=1}$) and 7×7 patches ($h^{w=7}$).

3.3 Color image texture classification

For color image texture classification, we consider the color textures of Outex13 [15]. This set contains 68 textures where every image has been divided into 20 non-overlapping sub-images each of size 128×128, thus providing a total of 1360 images, which have been evenly divided as training and test sets. We employ the morphological covariance as a texture descriptor [1]. Morphological covariance K' of an image f is defined as the volume Vol of the image (i.e., the sum of pixel values), eroded by a pair of points $P_{2,v}$ separated by a vector v (the SE is composed of only two pixels): $K'(f, P_{2,v}) = Vol(\epsilon_{P_{2,v}}(f))$. In practice K' is computed for varying length of v and the normalized version K is used for measurements: $K(f) = Vol(\epsilon_{P_{2,v}}(f))/Vol(f)$. The covariance based feature vector that describes a color image texture is computed using four directions $(0°, 45°, 90°, 135°)$ with distances ranging from 1 to 49 pixels in steps of size two. So, 25 values are available for the 4 directions on the 3 color channel making a vector of 300 values to describe an image. We also consider the extended MM covariance formulation of [4] that consider the concatenated results of three types of SEs (2 points, a cross, a square) within a 3×3 square, providing a vector of 900 values. With the obtained feature set, we considered two classifiers: a 1-nearest neighbor and a SVM. Figure 5 presents the classification accuracies. For classical covariance that uses SEs of only 2 points, the results are relatively close but better results are obtained with a learned complete lattice from colors. In this case there is no strong benefit in the use of patches. However, for the extended morphological covariance that uses more complex SEs, a significant gain appears: the learned complete lattice of colors is now much better than the classical lexicographic lattice and the patch-based learned lattice performs the best. The gain of using patches for MM texture classification appears now evident and confirms that the use of higher level cues than simple color in a MM texture feature extraction enables to obtain better features, and consequently better classification (whatever the considered classifier).

4 Conclusion

This paper has detailed an approach towards the construction of patch-based adaptive MM operators. The complete lattice of patches if learned by manifold learning from images and induces a patch-based learned complete lattice of the initial vectors of the image. To be efficient a three step strategy based on dictionary learning, manifold learning and out of sample extension has been devised. Patch-based adaptivity has been highlighted for morphological processing as an efficient way to preserve fine and repetitive structures for MM processing but also for image segmentation and texture classification.

References

1. Aptoula, E., Lefèvre, S.: A comparative study on multivariate mathematical morphology. Pattern Recognit. 40(11), 2914–2929 (2007)

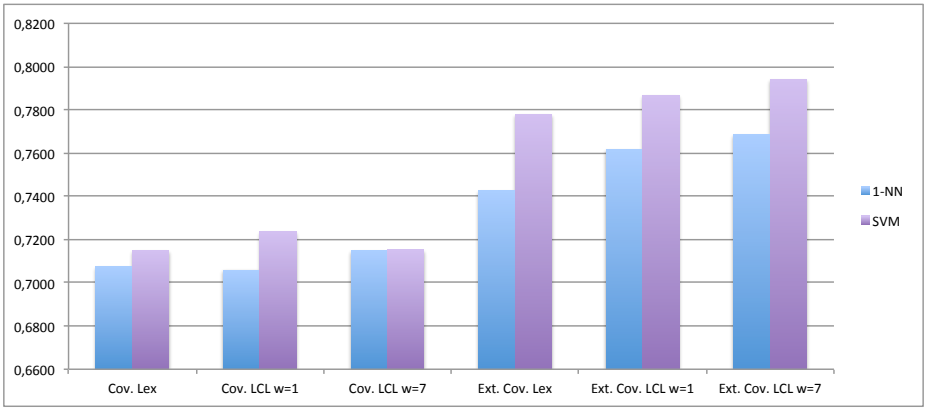

Fig. 5. Comparison between 1-Nearest Neighbor and SVM texture classification with Morphological covariances (Cov.) and extended covariances (Ext. Cov.) with lexicographic color lattice h^{Lex} or Learned Complete Lattice (LCL) from color ($h^{w=1}$) and 7×7 patches ($h^{w=7}$).

2. Aptoula, E., Lefèvre, S.: On lexicographical ordering in multivariate mathematical morphology. Pattern Recognition Letters 29(2), 109–118 (2008)
3. Aptoula, E., Lefèvre, S.: Multivariate mathematical morphology applied to colour image analysis. In: Collet, C., Chanussot, J., Chehdi, K. (eds.) Multivariate image processing: methods and applications, pp. 303–337. ISTE - John Wiley (2009)
4. Aptoula, E.: Extending morphological covariance. Pattern Recognition 45(12), 4524 – 4535 (2012)
5. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. 15(6), 1373–1396 (2003)
6. Buades, A., Coll, B., Morel, J.M.: Image denoising methods. a new nonlocal principle. SIAM Review 52(1), 113–147 (2010)
7. Chanussot, J., Lambert, P.: Bit mixing paradigm for multivalued morphological filters. In: International Conference on Image Processing and Its Applications. vol. 2, pp. 804 – 808 (1997)
8. Ćurić, V., Landström, A., Thurley, M.J., Hendriks, C.L.L.: Adaptive mathematical morphology – a survey of the field. Pattern Recognition Letters 47, 18 – 28 (2014)
9. Gersho, A., Gray, R.: Vector Quantization and Signal Compression. Kluwer Academic (1991)
10. Goutsias, J., Heijmans, H., Sivakumar, K.: Morphological operators for image sequences. Computer Vision and Image Understanding 62(3), 326–346 (1995)
11. Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer (2007)
12. Lezoray, O., Charrier, C., Elmoataz, A.: Rank transformation and manifold learning for multivariate mathematical morphology. In: EUSIPCO (European Signal Processing Conference). pp. 35–39 (2009)
13. Lézoray, O., Elmoataz, A.: Nonlocal and multivariate mathematical morphology. In: International Conference on Image Processing (IEEE). pp. 129–132 (2012)
14. Maragos, P., Vachier, C.: Overview of adaptive morphology: Trends and perspectives. In: Image Processing (ICIP), 2009 16th IEEE International Conference on. pp. 2241–2244 (Nov 2009)

15. Ojala, T., Maenpaa, T., Pietikainen, M., Viertola, J., Kyllonen, J., Huovinen, S.: Outex - new framework for empirical evaluation of texture analysis algorithms. In: Pattern Recognition, 2002. Proceedings. 16th International Conference on. vol. 1, pp. 701–706 vol.1 (2002)
16. Peyré, G.: Manifold models for signals and images. Computer Vision and Image Understanding 113(2), 249–260 (2009)
17. Roerdink, J.: Adaptivity and group invariance in mathematical morphology. In: Image Processing (ICIP), 2009 16th IEEE International Conference on. pp. 2253–2256 (Nov 2009)
18. Ronse, C.: Why mathematical morphology needs complete lattices. Signal Processing 21(2), 129–154 (1990)
19. Salembier, P.: Study on nonlocal morphological operators. In: International Conference on Image Processing (IEEE). pp. 2269–2272 (2009)
20. Ta, V.T., Elmoataz, A., Lezoray, O.: Nonlocal morphological levelings by partial difference equations over weighted graphs. In: ICPR (International Conference on Pattern Recognition). CD Proceedings (2008)
21. Ta, V.T., Elmoataz, A., Lezoray, O.: Partial difference equations over graphs: Morphological processing of arbitrary discrete data. In: European Conference on Computer Vision. vol. LNCS 5304, pp. 668–680 (2008)
22. Ta, V., Elmoataz, A., Lézoray, O.: Nonlocal pdes-based morphology on weighted graphs for image and data processing. IEEE Transactions on Image Processing 20(6), 1504–1516 (June 2011)
23. Talwalkar, A., Kumar, S., Mohri, M., Rowley, H.A.: Large-scale SVD and manifold learning. Journal of Machine Learning Research 14(1), 3129–3152 (2013)
24. Velasco-Forero, S., Angulo, J.: Mathematical morphology for vector images using statistical depth. In: Soille, P., Pesaresi, M., Ouzounis, G. (eds.) Mathematical Morphology and Its Applications to Image and Signal Processing, Lecture Notes in Computer Science, vol. 6671, pp. 355–366. Springer Berlin Heidelberg (2011)
25. Velasco-Forero, S., Angulo, J.: On nonlocal mathematical morphology. In: Hendriks, C., Borgefors, G., Strand, R. (eds.) Mathematical Morphology and Its Applications to Signal and Image Processing, Lecture Notes in Computer Science, vol. 7883, pp. 219–230. Springer Berlin Heidelberg (2013)
26. Yang, S., Li, J.X.: Spatial-variant morphological filters with nonlocal-patch-distance-based amoeba kernel for image denoising. Image Analysis and Stereology 34(1), 63–72 (2015)