Graph-based ordering scheme for color image filtering

Olivier Lezoray, Cyril Meurie, Abderrahim Elmoataz

International Journal of Image and Graphics

Graph-based ordering scheme for color image filtering

O. LEZORAY[1]*
C. MEURIE[2]
A. ELMOATAZ[1]
[1] *Université de Caen Basse-Normandie and ENSICAEN,*
GREYC UMR CNRS 6072, 6 Bd. Maréchal Juin, F-14050 CAEN, FRANCE
[2] *INRETS-LEOST, 20 rue Elisee Reclus, BP 317, 59666 Villeneuve d'ascq, FRANCE*
olivier.lezoray@unicaen.fr, abder@greyc.ensicaen.fr, cyril.meurie@inrets.fr

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

This paper presents a graph-based ordering scheme of color vectors. A complete graph is defined over a filter window and its structure is analyzed to construct an ordering of color vectors. This graph-based ordering is constructed by finding a Hamiltonian path across the color vectors of a filter window by a two-step algorithm. The first step extracts, by decimating a minimum spanning tree, the extreme values of the color set. These extreme values are considered as the infimum and the supremum of the set of color vectors. The second step builds an ordering by constructing a Hamiltonian path among the vectors of color vectors, starting from the infimum and ending at the supremum. The properties of the proposed graph-based ordering of vectors are detailed. Several experiments are conducted to assess its filtering abilities for morphological and median filtering.

Keywords: Ordering of vectors, color, filtering, mathematical morphology, median, graph, Hamiltonian path.

1. Introduction

Color image processing is a very important area of research which belongs to the processing of multichannel images[20,28,29]. A large number of nonlinear image filters have been initially developed for the processing of gray-scale images. It is now admitted that a direct extension of these algorithms to color images by a component-wise approach is not suitable since it neglects the correlation that exists between the color channels of a natural image. Moreover, this can produce color shifts or other artifacts[16]. To address this problem, a vectorial processing of color images is preferable.

Nonlinear filters such as median or morphological filters require the pixels contained in a filter window to be ordered. Since there is no natural ordering for vector

*Corresponding author.

data, their extension to color images is not straightforward. The definition of an ordering of color vectors is therefore of high interest for processing color images[20,23] since it can be used for image filtering, edge detection, spatial interpolation, etc[20].

In this paper, a new graph-based approach to the ordering of color vectors is presented. A complete graph is defined over a filter window and its structure is analyzed to construct an ordering of color vectors. This graph-based ordering is constructed by finding a Hamiltonian path[8] across the color vectors of a filter window by a two-step algorithm. The first step extracts, by decimating a minimum spanning tree, the extreme values of the color set. These extreme values are considered as the infimum and the supremum of the set of color vectors. The second step builds an ordering by constructing a Hamiltonian path among the vectors of color vectors, starting from the infimum and ending at the supremum.

The rest of this paper is organized as follows. In the next section, an overview of multivariate ordering of vectors is presented and various schemes for ordering vectors are described. In Section 3, these schemes are compared according to their abilities for color morphological or median filtering. In Section 4, after having recalled basic definitions on weighted graphs, the proposed graph-based ordering of color vectors is described. Results and discussion are presented in Section 5. In the final section, some concluding remarks are made.

2. Multivariate ordering of vectors

To process multivariate data such as color vectors, the three main multivariate orderings of vectors are the pre-order, the partial order and the total order. Before the description of these orderings, we provide definitions of the relations used to characterize an ordering of vectors[10,40,17].

Definition 1. Let R be a binary relation on a given set A.

- R is reflexive if $\forall \mathbf{x} \in A, \mathbf{x}R\mathbf{x}$,
- R is transitive if $\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in A, \mathbf{x}R\mathbf{y}$ and $\mathbf{y}R\mathbf{z} \Rightarrow \mathbf{x}R\mathbf{z}$,
- R is anti-symmetric if $\forall \mathbf{x}, \mathbf{y} \in A, \mathbf{x}R\mathbf{y}$ and $\mathbf{y}R\mathbf{x} \Rightarrow \mathbf{x} = \mathbf{y}$.

Definition 2. A binary relation R on a set A is a pre-order if R is reflexive and transitive.

Definition 3. A binary relation R on a set A is a partial order if R is reflexive, transitive and anti-symmetric.

Definition 4. A binary relation R on a set A is a total order if R is a partial order and if $\forall \mathbf{x}, \mathbf{y} \in A, \mathbf{x}R\mathbf{y}$ or $\mathbf{y}R\mathbf{x}$.

Therefore, a complete ordering on a set A is an ordering where any pair of vectors can be ordered (e.g. the binary relation $\leq$ on $\mathbb{R}$).
A multivariate image can be represented by the mapping $\mathbb{Z}^l \to \mathbb{R}^p$ where l is the image dimension and p the number of channels. Let $W = \{\mathbf{x}_k \in \mathbb{Z}^l; k = 1, 2, \ldots, N\}$

design a filter window of finite length N where $\mathbf{x}_1$ determines the position of the filter window (i.e. $\mathbf{x}_1$ occupies the central location in the filter window). From W one therefore gets a set $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ of N p-dimensional vectors: $\mathbf{x}_i = \{\mathbf{x}_i^1, \mathbf{x}_i^2, \cdots, \mathbf{x}_i^p\}, \mathbf{x}_i \in \mathbb{R}^p$. A classical way to define an ordering relation between vectors is to use a transform[9] h from $\mathbb{R}^p$ into $\mathbb{R}^q$ followed by the natural ordering on each dimension of $\mathbb{R}^q$. With $h : \mathbb{R}^p \to \mathbb{R}^q$, and $\mathbf{x} \to h(\mathbf{x})$ then $\forall(\mathbf{x}_i, \mathbf{x}_j) \in \mathbb{R}^p \times \mathbb{R}^p, \mathbf{x}_i \leq \mathbf{x}_j \Leftrightarrow h(\mathbf{x}_i) \leq h(\mathbf{x}_j)$. When h is bijective, this corresponds to defining a space filling curve that goes through each point of the $\mathbb{R}^p$ space just once and thus induces a total ordering. According to these definitions, we can review some several possible types of multivariate orderings of vectors as it has been done by Barnett[5]. The ones we present are the marginal, the reduced and the conditional orderings. We recall their principles here.

- In the marginal ordering, vectors are ordered in each dimension independently ($q = p, h = Identity$). This order is a partial ordering and it is now admitted that this approach is not satisfactory since it can produce new vectors which do not belong to the initial set.
- In the reduced ordering, $q = 1$ and h denotes a chosen distance metric which differentiates the possible reduced ordering schemes. Each vector is reduced to a scalar and the vector data are sorted according to the obtained scalar values. A classical way to define the h transform is to use a cumulative distance. This type of ordering is very popular and is used for color vector median filters for instance[20,21]. This order is a pre-order.
- In the conditional (or lexicographic) ordering, the vectors are ordered according to a hierarchical order of the components. Therefore, $q = 1$ and $h : \mathbb{R}^p \to \mathbb{R}, \mathbf{x}_i \to \mathbf{x}_i^k$ with k adaptively determined. For two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, one has:

$$\mathbf{x}_i \leq \mathbf{x}_j \begin{cases} \mathbf{x}_i^1 < \mathbf{x}_j^1 \text{ , or} \\ \mathbf{x}_i^1 = \mathbf{x}_j^1 \text{ , and, } \mathbf{x}_i^2 < \mathbf{x}_j^2 \text{ or} \cdots \\ \mathbf{x}_i^1 = \mathbf{x}_j^1 \text{ , and, } \mathbf{x}_i^2 = \mathbf{x}_j^2 \cdots \mathbf{x}_i^p < \mathbf{x}_j^p \end{cases}$$

  This ordering is a total ordering of vectors but it introduces a strong dissymmetry between the components.

Since processing multivariate data is of interest in many research areas (e.g. color or multi spectral image processing), a wide range of different schemes for ordering vectors can be found in literature [2,4,6,7,11,12,14,18,25,26,27,28,35,38,39]. Most of these works rely on modified lexicographic ordering schemes since this is the only true total ordering relation commonly used in literature. However, another total ordering of color vectors, called the bit-mixing ordering, has been proposed by Chanussot[6]. The bit-mixing is based on the binary representation of each component of the considered vector $\mathbf{x}$. If the p components of $\mathbf{x}$ are coded with b bits each, the $p \cdot b$ available bits are blended together to build the $p \cdot b$ bits long scalar value $h(\mathbf{x})$[6,15].

The considered transform h can then be written as follows:

$$h(\mathbf{x}) = \sum_{k=1}^{b} \left\{ 2^{p\cdot(b-k)} \sum_{i=1}^{p} 2^{(p-i)} \mathbf{x}^{i \triangleright k} \right\}$$

where $\mathbf{x}^{i \triangleright k}$ denotes the k^{th} bit of the i^{th} component of $\mathbf{x}$. In the case of color images, $p = 3$, $b = 8$.
In the context of space filling-curves, it is worth mentioning the works of Regazonni [30,34] where a given curve is defined for median filtering. We do not consider that issue of constructing a space filling curve on all the space $\mathbb{R}^p$. We will see in Section 4.2 that an equivalent notion (the Hamiltonian path) can be used to construct an ordering of color vectors.

3. Ordering and filtering

One very important field of application of multivariate image processing and in particular color images is filtering[28,6,20,29]. In this section, we focus on the link between ordering color vectors and filtering (morphological[31,9] or median[23,6]). Morphological or median filters are usually based on specific orderings of color vectors which have advantages and disadvantages. We give details about both in the sequel.

3.1. *Vector morphological filters*

Mathematical morphology is a nonlinear approach to image processing which relies on a fundamental structure, the complete lattice $\mathcal{L}$. A complete lattice is defined such that[31]:

- An ordering relation $\leq$ is defined over $\mathcal{L}$,
- For every finite subset $\mathcal{K}$ of $\mathcal{L}$, there exists a supremum $\vee\mathcal{K}$ and an infimum $\wedge\mathcal{K}$.

A marginal approach can be used, but a purely vectorial approach is preferable. Indeed, with a marginal approach, the supremum and the infimum do not always belong to the lattice and false colors can appear[6]. Another constraint to the definition of morphological filters is therefore to impose that the supremum and the infimum of a given set do belong to this one. Total orderings are thus usually considered for color morphological operators. In this frame, the main orderings of color vectors are the lexicographic ordering[11,2] and the bit-mixing ordering[6] which are total orderings and which both fulfill all the requirements of the complete lattice $\mathcal{L}$.

The lexicographic ordering is however strongly dissymmetric and most of the ordering of color vectors decisions are taken on the level of the first component which implies the attribution of a priority to the components. This provides operators the behavior of which is not homogeneous in a color space. The choice of the priority component being difficult[3,25,1], this can be alleviated by considering perceptual color spaces based on Luminance/Hue/Saturation[2,18] where the ordering is more natural from a human perception point of view. Even if this ordering has drawbacks, this is the most commonly used and it has been studied by several authors for morphological filtering[12,18,26,27,35,38,39].

The bit-mixing ordering enables to limit the dissymmetry between the components. However, since it is based on an interlacing of bits, it is reduced to operate in color spaces where components are described by integers, which is not the case of a lot of color spaces. The bit-mixing ordering was therefore mainly conceived to operate in the RGB color space.

Once an ordering of color vectors is defined, one can apply the two main morphological operations that is to say the erosion ϵ and the dilatation δ. Regardless the ordering scheme, the color erosion (ϵ) and dilatation (δ) over a filter window W centered on $\mathbf{x}$ are given by:

$$\epsilon(\mathbf{x}) = \{\mathbf{y} : \mathbf{y} = \wedge W\} \text{ and } \delta(\mathbf{x}) = \{\mathbf{y} : \mathbf{y} = \vee W\}$$

where $\vee$ and $\wedge$ are respectively the supremum and the infimum of a set. Other morphological operators can be obtained by composition of these elementary operations: the opening ($\gamma = \delta\epsilon$) and the closing ($\varphi = \epsilon\delta$) for instance.

3.2. *Vector Median Filters*

The most popular vector filter is the Vector Median Filter (VMF)[21] introduced by Astola[4]. We do not detail the state of the art on vector median filters since it is out of the scope of this paper and excellent reviews can be found in[20,21,23,29]. Such vector filters are based on the ordering of vectors in a filter window W. Each vector $\mathbf{x}_k \in W$ is associated with a distance measure:

$$R_k = \sum_{i=1}^{N} \|\mathbf{x}_k - \mathbf{x}_i\|_\gamma \text{ for } k = 1, 2, \ldots, N$$

where $\|\mathbf{x}_k - \mathbf{x}_i\|_\gamma$ quantifies the Minkowski distance between two vectors $\|\mathbf{x}_k - \mathbf{x}_j\|_\gamma = \left(\sum_{i=1}^{p} \left|\mathbf{x}_k^i - \mathbf{x}_j^i\right|^\gamma \right)^{\frac{1}{\gamma}}$ where γ designs the chosen norm which is usually the Euclidean one ($\gamma = 2$). The VMF output is the sample $\mathbf{x}_{(1)}$ associated with the minimal aggregated distance:

$$\mathbf{x}_{(1)} = arg \min_{\mathbf{x}_i \in W} R_i$$

According to the distance between two vectors, it is possible to differentiate the techniques operating on the vector distance domain[21], the angular domain[37] or their combinations[13].

3.3. *Discussion*

The first observation is that reduced orderings used for vector median filters are not adapted to morphological filtering since they do not define a complete lattice. They are however adapted to any type of multivariate image and to any color space. The second observation is that the ordering schemes used for morphological processing are usually total orderings, but they have the main drawback of being suited to only one given color space: RGB for the bit-mixing ordering and $IHSL$[11] for the lexicographic ordering. Moreover, these ordering schemes cannot be used to perform morphological operations on multivariate images. For the bit-mixing ordering, we are limited to integers and to the number of bits available to code an integer. For the lexicographic ordering, the dominant role of the first component is difficult to overcome except in a perceptual color space such as $IHSL$. The final observation is that all the usual orderings of vectors have disadvantages; we would like to have one ordering of vectors the basis of which is enough general to permit morphological and median filtering on vectors of arbitrary dimensions.

4. Graph-based Ordering

In this section, we propose a graph-based alternative to the classical approaches to ordering the vectors of color vectors.

4.1. *Preliminaries on graphs*

We provide some basic definitions on graph theory[8]. A graph $\mathcal{G}$ is a couple $\mathcal{G} = (V, E)$ where V is a finite set of vertices and E is a subset $E \subseteq V \times V$. The elements of V are the vertices of the graph and the elements of E are the edges of the graph. Two vertices u and v in a graph are adjacent if the edge (u, v) exists in E, the two vertices being then called neighbor vertices.

The degree $\delta(v)$ of a vertex v is the number of edges incident to the vertex. $\delta : V \to \mathbb{N}$ is defined as $\delta(v) = |u \sim v|$ where $|\cdot|$ denotes the cardinal of a set. The relation $u \sim v$ denotes the set of vertices u connected to the vertex v via the edges $(u, v) \in E$: u is a neighbor adjacent vertex of v. If a vertex has a unity degree, it is called a leaf.

A path p is a set of vertices $p = (v_1, v_2, \cdots, v_k)$ such as there is an edge for each two successive vertices of the path: $\forall i \in [1, k[$, the edge $(v_i, v_{i+1}) \in E$. The length of a path corresponds to its number of edges. A path is simple if an edge is covered only once. A path is Hamiltonian if it uses all the vertices exactly once (this problem is NP-complete). A complete graph is a graph where an edge connects every pair of vertices. A complete graph with n vertices has $n(n-1)/2$ edges and the degree of each vertex is $(n-1)$.

A graph is connected when for every pair of vertices u and v there is a path in which $v_1 = u$ and $v_k = v$. A graph is undirected when the set of edges is symmetric, i.e., for each edge $(u, v) \in E$, we have also $(v, u) \in E$. In the rest of this paper, we consider only simple graphs for which maximum one edge can link two vertices.

These simple graphs are always assumed to be connected and undirected[8]. A graph, as defined above, is said to be weighted if it is associated with a weight function $w : E \to \mathbb{R}^+$ satisfying $w(u,v) > 0$ if $(u,v) \in E$, $w(u,v) = 0$ if $(u,v) \notin E$ and $w(u,v) = w(v,u)$ for all edges in E since we consider undirected graphs.

We can now define the space of functions on graphs. Let $\mathcal{H}(V)$ denote the Hilbert space of real-valued functions on vertices, in which each $f : V \to \mathbb{R}^+$ assigns a real value $f(v)$ to each vertex v. A function f in $\mathcal{H}(V)$ can be thought as a column vector in $\mathbb{R}^{|V|}$. Similarly, one can define $\mathcal{H}(E)$ the space of real-valued functions on edges, in which each one $g : E \to \mathbb{R}^+$ assigns a real value to each edge e.

A tree is a connected acyclic simple graph. A spanning tree of a connected, undirected graph $\mathcal{G}$ is a tree composed of all the vertices and some of the edges of $\mathcal{G}$. A minimum spanning tree (MST) is then a spanning tree with weight less than the weight of every other spanning tree. Therefore, a minimum spanning tree $\mathcal{T}(\mathcal{G})$ of a graph $\mathcal{G}$ is a weighted connected graph $\mathcal{T}(\mathcal{G}) = (V', E')$ where the sum of the weights $\sum\limits_{(u,v)\in E'} w(u,v)$ is minimum. For a graph $\mathcal{G}$ of n vertices, its MST $\mathcal{T}(\mathcal{G})$ has exactly $(n-1)$ edges. The MST can be efficiently computed in $O(|E|log|V|)$ using Prim's algorithm with appropriate data structures.

4.2. *Graphs and Ordering of vectors*

As previously mentioned, there is an equivalence:
(total ordering on $\mathbb{R}^p$)$\Leftrightarrow$(bijective application $h : \mathbb{R}^p \to \mathbb{R}$)$\Leftrightarrow$(space filling curve in $\mathbb{R}^p$)
We recall that a space filling curve is a curve that goes through each point of the space just once. Therefore, if the space is represented by a connected graph, we also have the equivalence: (space filling curve in $\mathbb{R}^p$)$\Leftrightarrow$(Hamiltonian path on $\mathbb{R}^p$). Usual definitions of space filling curves[34,30] in $\mathbb{R}^p$ use curves (e.g. a Peano scan) which are independent of the image spatial structure and therefore they do not respect topology. Indeed, finding an optimal Hamiltonian path is too difficult (NP-complete) to be directly solved on $\mathbb{R}^p$: for a graph of n vertices, there are $(n-1)!$ possible Hamiltonian paths. In this paper, we take a Hamiltonian path point of view of the ordering of vectors. However, we propose to dynamically construct such a Hamiltonian path on a filter window W rather on the complete space $\mathbb{R}^p$.

To a given filter window W, we can associate a complete graph the vertices of which correspond to the vectors of W. This corresponds to a function $f \in \mathcal{H}(V)$, $f : V \to W$ which associates a color vector $\mathbf{x} \in W$ to each vertex. Similarly, we can associate a weight to each edge of the graph, $w \in \mathcal{H}(E), w : E \to \mathbb{R}^+$. Classically, we consider $w(u,v) = \|f(u) - f(v)\|_2$.

In this paper, we just focus on color images ($p = 3$) but the principle remains the same for higher dimensions. To illustrate our approach, we use a reference image (figure 1, a painting by Joan Miro called "The singer") in color mathematical morphology which was introduced by Hanbury[2,11,17]. In the sequel, we illustrate our new graph-based ordering approach of color vectors on a predefined filter window W of this image (specified by the zoomed and surrounded area in the top right of figure 1).

Fig. 1. Test image for morphological filtering operations.

4.3. *Inf and Sup extraction*

As previously mentioned, to define a Hamiltonian path over a filter window W, we consider the complete graph $\mathcal{G}_0$ over W. Since it is difficult to find an optimal Hamiltonian path on $\mathcal{G}_0$ among all the $(|W|-1)!$ different possibilities, we propose to approximate this path. Instead of trying to directly define the Hamiltonian path, we begin by extracting its bounds ($\wedge$ and $\vee$).

Let $\mathcal{T}_0 = \mathcal{T}(\mathcal{G}_0)$ denote the MST of $\mathcal{G}_0$. An MST being a generalization to higher dimensions of a one-dimension sorted list[36], we can use its structure to find candidate bounds of the Hamiltonian path. A vertex v of a Hamiltonian path is one of its bounds if $\delta(v) = 1$: it is a leaf. We use this principle to extract them. Let $\mathcal{N}_0 = \{u|\delta(u) = 1, u \in \mathcal{T}_0\}$ denote the leaves of $\mathcal{T}_0$. The vertices in $\mathcal{N}_0$ are the only candidates for bounds of the Hamiltonian path. Since most of the time $|\mathcal{N}_0| > 2$, $\mathcal{N}_0$ has to be reduced to only two elements. To that aim, we iterate the same process on the complete graph constructed over the vertices of $\mathcal{N}_0$ until $|\mathcal{N}_i| = 2$ with i the iteration number.

To sum it up, to extract the bounds of a Hamiltonian path, the principle can be described as follows:

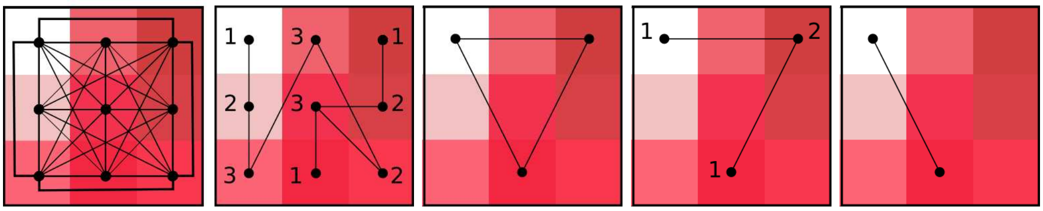
$i \leftarrow 0$
Construct the complete graph $\mathcal{G}_i$ over the filter window W
Repeat
  $\mathcal{T}_i = \mathcal{T}(\mathcal{G}_i)$
  $\mathcal{N}_i = \{u|\delta(u) = 1, u \in \mathcal{T}_i\}$
  Construct the complete graph $\mathcal{G}_{i+1}$ over the vertices of $\mathcal{N}_i$
  $i \leftarrow i + 1$
until $(|\mathcal{N}_{i-1}| = 2)$
$\mathcal{T}_i = \mathcal{G}_i$
$\mathcal{N}_i = \{u \in \mathcal{T}_i\}$

At the end of the process, $\mathcal{N}_i = \{u, v\}$ contains two vertices considered as bounds of the Hamiltonian path. However, one still has to define which one of these two vertices is the $\vee$ (respectively the $\wedge$). The $\wedge$ is identified as the closest vertice to a reference color $\mathbf{x}_{ref}$ [11,32] which is usually black:

$$\vee = arg \max_{v \in \mathcal{N}_i} \|f(v), \mathbf{x}_{ref}\|_2 \text{ and } \wedge = arg \min_{v \in \mathcal{N}_i} \|f(v), \mathbf{x}_{ref}\|_2$$

The use of a reference color is not new and was proposed by Soria-Frisch in[32] and Hanbury for the $IHSL$ color space[11].

Computing the MST of a graph is $O(|E|log|V|)$ and $O(N^2 logN)$ on a complete graph since $|V| = |W| = N$. One then iterates the process on the complete graph built from the leaves of the MST. There is no bound on the number of leaves of an MST for $p > 2$ (see in [33]), we therefore have determined an experimental upper bound of this number. For a complete graph built over a filter window of size N, an upper bound of the number of leaves of the MST is $\lceil N/2 \rceil$ where $\lceil \cdot \rceil$ denotes the smallest integer upper than the operand. This means that, at each iteration, the number of surviving vertices is divided by two (this is an upper bound). One then can prove that the complexity of this step is $O(\frac{N^3}{2} logN)$.



(a) $\mathcal{G}_0$ (b) $\mathcal{T}_0$ (c) $\mathcal{G}_1$ (d) $\mathcal{T}_1$ (e) $\mathcal{G}_2$

Fig. 2. The graphs $\mathcal{G}_i$ and $\mathcal{T}_i$ in the different steps of the algorithm for Hamiltonian path bounds extraction. For each $\mathcal{T}_i$, the vertices degrees are superimposed. $\mathcal{T}_2$ is not shown here.

Figure 2 presents the different steps of the algorithm. The complete graph $\mathcal{G}_0$ is constructed over the filter window W (Figure 2(a)) and its MST $\mathcal{T}_0$ is computed (Figure 2(b)). A new complete graph $\mathcal{G}_1$ Figure 2(c)) is constructed over the leaves of $\mathcal{T}_0$ and its MST $\mathcal{T}_1$ is computed (Figure 2(d)). Finally, A complete graph $\mathcal{G}_2$ of two vertices is obtained (Figure 2(e)). With $\mathbf{x}_{ref}$ as black, the $\vee$ is the top left pixel of W and $\wedge$ the bottom middle one.

4.4. *Ordering of vectors construction*

Once the two bounds of the Hamiltonian path have been determined, the complete Hamiltonian path can be constructed. On the filter window W under consideration, a Hamiltonian path $p = (v_1, v_2, \cdots, v_k)$ must respect the following properties:

- The length of p is $|W| = N$,
- Its bounds are leaves i.e. $\delta(v_1) = \delta(v_k) = 1$,
- $v_1 = \wedge$ and $v_k = \vee$,
- The other vertices have a degree of 2: $\forall v \in \{v_2, \cdots, v_{k-1}\}, \delta(v) = 2$.

The construction of the Hamiltonian path we propose is based on the nearest neighbor principle on the initial complete graph $\mathcal{G}_0$. To determine the nearest neighbor u of a vertex v we consider the weight of the edge $w(u, v)$ but also the saliency of the neighbor u. The saliency of a vertex quantifies its global importance in the set of MST which where generated during the inf and sup extraction. It is defined as follows:

$$\Delta(v) = \sum_{i=0}^{i_{max}} (i+1) \cdot \delta(\mathcal{T}_i, v) \text{ where } \delta(\mathcal{T}_i, v) = \delta(v), v \in \mathcal{T}_i$$

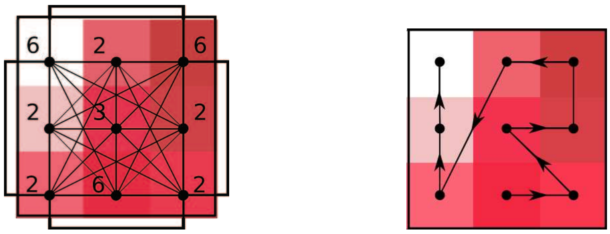
where i corresponds to an iteration number in the Hamiltonian path bounds extraction and i_{max} the total number of iterations. The bounds v_1 and v_N have the highest saliences. Otherwise, the saliency of a vertex is all the more important when it survives in the successive $\mathcal{T}_i$. For instance, the saliency of the top left pixel v_j of the filter window in figure 2(a) is

$$\Delta(v_j) = 1 \cdot \delta(\mathcal{T}_0, v_j) + 2 \cdot \delta(\mathcal{T}_1, v_j) + 3 \cdot \delta(\mathcal{T}_2, v_j) = 1 \cdot 1 + 2 \cdot 1 + 3 \cdot 1 = 6$$

The construction of the Hamiltonian path is $O(N^2)$ and can then be summarized as follows:

$$\begin{cases} v_1 = \wedge \text{ and } v_N = \vee \\ v_{j+1} = arg \min\limits_{\substack{u \sim v_j \\ u \notin \{v_1, \cdots, v_j\}}} (w_{uv_j} \Delta(u)) \quad \forall j = 1, \cdots, (N-2) \end{cases}$$

Figure 3 illustrates the construction of the ordering of vectors of W as a Hamiltonian path: on the complete graph $\mathcal{G}_0$ (Figure 3(a), with vertices saliences superimposed), one obtains the path depicted by figure 3(b).



(a) $\mathcal{G}_0$

(b) The constructed Hamiltonian path.

Fig. 3. (a) The graph $\mathcal{G}_0$ with vertices saliences $\Delta(v)$ superimposed and (b) the constructed Hamiltonian path.

5. Evaluation and discussion

In this section, we evaluate the four orderings of vectors detailed in the previous sections, namely the reduced ordering based on distances, the lexicographic ordering, the bit mixing ordering and the proposed graph-based ordering.

5.1. *Ordering schemes comparison*

First, to see the influence of each ordering scheme, we apply each of them on the filter window W of figure 1. Results are shown in figure 4. The first row presents the filter window W. Other rows show the results for the different orderings of vectors. As it was previously noticed, the marginal approach creates some color artifacts

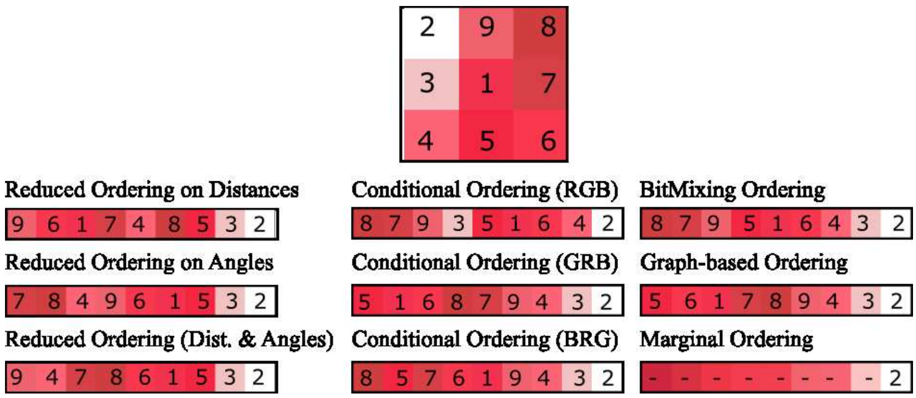


Fig. 4. Comparison of several schemes for ordering vectors on a filter window (from figure 1). Each color is indexed in the filter window (top row). These indexes are superimposed for all the ordering schemes (- indicates that the color vector does not belong to the filter window).

(appearance of false colors). One can however note that this is not visually obvious.

For the lexicographic ordering, several permutations of the color channels have been studied to assess the dominant role of the first channel. In this case, considering the Green channel as dominant provides better results but this might not be the case for another filter window. For the reduced ordering, the chosen distance between vectors has high influence on the final ordering: some visually close color can be far away one from the other in the ordering. The bit-mixing ordering is very close to the lexicographic RGB ordering which assesses the fact that the way the bits are mixed still involves dissimetry. The graph-based ordering is visually satisfying and very close to the lexicographic GRB ordering (only two permutations).

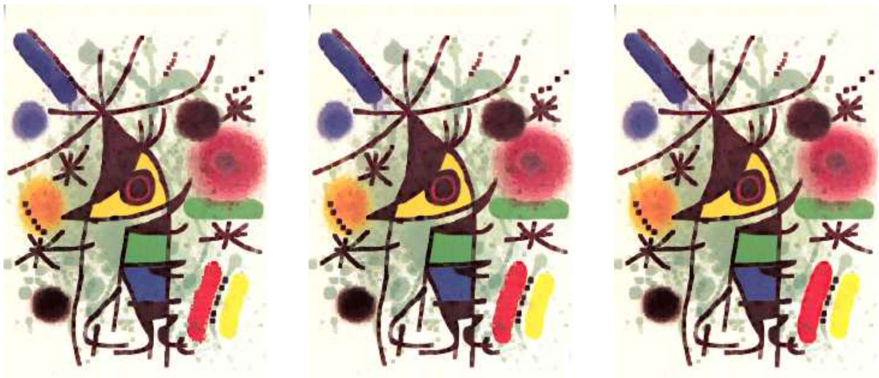
We now analyze the computational complexity of the different ordering schemes for a given filtering window W of size N to obtain a complete ordering of the vectors. With usual sorting algorithms, the lexicographic ordering computational complexity is $O(N^2)$ and the bit-mixing ordering computational complexity is $O(N^2)$. The reduced ordering based on distances computational complexity is $O(N^2)$. The graph-based ordering computational complexity is $O(\frac{N^3}{2} logN)$.

5.2. *Morphological Filtering*

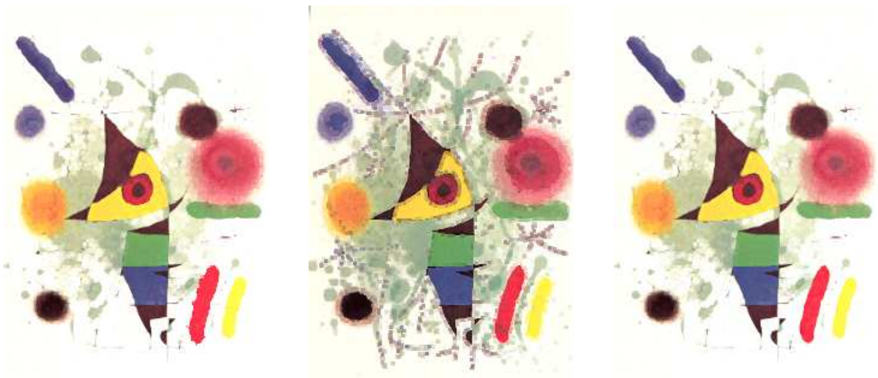
In this subsection, we only consider the lexicographic, the bit mixing and the graph-based orderings for morphological filtering. Figure 5 presents a comparison of the considered orderings of vectors for the two main morphological operations: the erosion (ϵ) and the dilatation (δ).

The considered color space is RGB, the filter window is a 3×3 square and the reference color for the graph-based ordering is black. In figure 5, few differences can be noticed between the three orderings of vectors. For the erosion the results look similar for the three orderings except for the lexicographic ordering in the red areas of the image. The disadvantages of the lexicographic ordering which gives a dominant role to the first component is well observed in red areas of figure 5(a). The same observation can be made for the lexicographic dilatation in figure 5(d). The bit mixing ordering performs good for the erosion but it is less good than the other two orderings for the dilatation: the bit mixing is very sensitive to the small color variations in the background of the image.

The reference color used in the graph-based ordering has an influence on the determination of the $\vee$ and the $\wedge$. Usually one will take black as a reference color, however for some special considerations, it might be interesting to use another color. We have performed an erosion in the RGB color space with different reference colors: black, red, green, blue and yellow. All the results are presented in figure 6(a)-(e), each column of the first row corresponds to a reference color. The first row gives an erosion of figure 1 with the five different reference colors. To illustrate the differences between the obtained results, we first computed the difference between the original image and black erosion (figure 6(f)). This gives an idea of the modifications made by the erosion in the image. To see the influence of the reference color, the differences between black erosion and the other reference color erosions



(a) ϵ Lexicographic (b) ϵ Bit Mixing (c) ϵ Graph



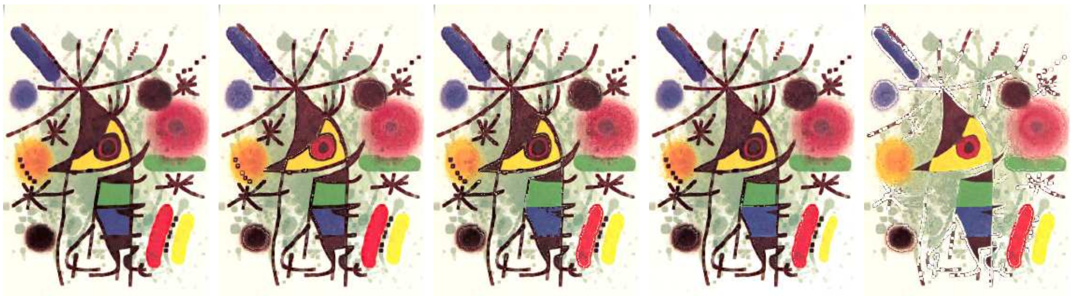
(d) δ Lexicographic (e) δ Bit Mixing (f) δ Graph

Fig. 5. ϵ and δ with different orderings on the figure 1.

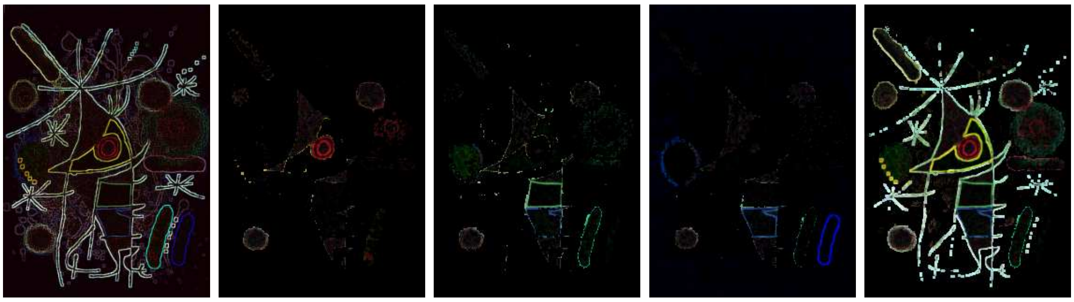
are computed. The differences between the obtained erosions are given in figure 6(g)-(j). As expected, changing the reference color changes the way the erosion acts on some colors since it favors the colors close to the reference: look at the eye of the singer for the red erosion and the yellow elongated pattern (bottom right) for the blue erosion. This becomes quite clear for the result of yellow erosion: the majority of the colors of the image are closer to yellow than to black and this last color tends to disappear in yellow erosion.

5.3. *Median Filtering*

In this subsection, we consider the abilities of the different orderings for median filtering. The aim here is not to conceive an efficient noise filtering operator but to study the influence of the ordering on median filtering. For comparison, the reduced ordering based on distances is considered as the reference. This enables us



(a) Black (b) Red (c) Green (d) Blue (e) Yellow



(f) Black (g) Red (h) Green (i) Blue (j) Yellow

Fig. 6. Influence of the reference color on the graph ordering for an erosion.

to perform filtering with VMF, BVDF or DDF operators. To compare the different orderings of vectors, a standard image is considered and is presented in figure 7 (similar results were obtained on other standard images). The experiments were conducted in the RGB color space with three different orderings: reduced ordering based on distances, graph ordering and bit mixing ordering. The original test images have been corrupted by impulse noise expressed as[19]:

$$x_{i,j} = \begin{cases} v & \text{with probability } p_v \\ o_{i,j} & \text{with probability } 1 - p_v \end{cases}$$

where i, j characterize the sample position, $o_{i,j}$ is the original sample, $x_{i,j}$ represents the sample from the noisy image, p_v is a corruption probability and $v = (v_R, v_G, v_B)$ is a noise vector of intensity random values. For the experiments, the considered degree of the impulse noise corruption p_v has ranged from 0% to 30%. To evaluate the achieved results, objective criteria as Mean Absolute Error (MAE), Mean Square Error (MSE) and Normalized Color Differece (NCD) have been used[19].

Figure 8 presents the results of the first conducted experimentation with a VMF filtering for the different orderings. Each figure presents an objective criterion according to the level of noise corrupting the image. It is easy to see that the orderings of vectors can be classified in the following way according to their impulse

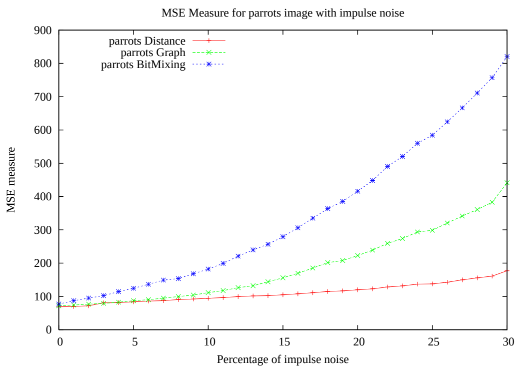
Fig. 7. Test color image.

noise median filtering abilities: reduced ordering based on distances, graph ordering, bit mixing ordering. The bit mixing is the least suitable ordering and the graph ordering has a behavior similar to reduced ordering for small quantities of impulse noise.

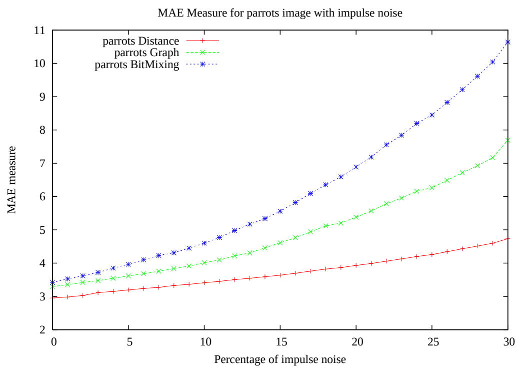
We have led the same experiments with a BVDF filtering for the reduced ordering based on distances and the graph ordering and we got the same type of curves (not shown here for the sake of brevity) than for the VMF assessing the superiority of the reduced ordering. It is worth noting that to obtain a BVDF with the graph ordering, one has only to change the weighting of the edges between the vertices of the graph: $w(u, v) = \theta(f(u), (v))$ where $\theta(\mathbf{x}_i, \mathbf{x}_j)$ represents the angle between the two color vectors $\mathbf{x}_i$ and $\mathbf{x}_j$. We then compare the DDF filtering scheme with a reduced ordering and a graph ordering $(w(u, v) = \theta(f(u), (v)) \left\| f(u) - f(v) \right\|_2)$. Results are shown in figure 9. One can see on all the curves that, for the DDF filtering, the graph ordering always provides better results than the reduced ordering. This comes to the conclusion that the graph ordering we propose in this paper is suitable for median filtering but the weights of the edges in the initial graph have to be carefully determined. This brings our graph ordering approach closer to the problem of the determination of weights in weighted median filtering[24,22].

To give a visual illustration of the differences between the different orderings of vectors, figure 10 presents a zoomed area of the Parrots image (figure 7(c)). The zoomed area (figure 10(e)) has been corrupted by 15% of impulse noise (figure 10(a)) and filtered first by a VMF with reduced ordering (figure 10(b)), graph ordering (figure 10(c)), or bit mixing ordering (figure 10(d)) and then by a DDF with reduced ordering (figure 10(f)) or graph ordering (figure 10(g)). The best visual results, as expected from the curves in figures 8 and 9, are obtained by the VMF with reduced ordering. One can also notice the better performance of graph ordering while using both distance and direction for weighting the edges of the graph.

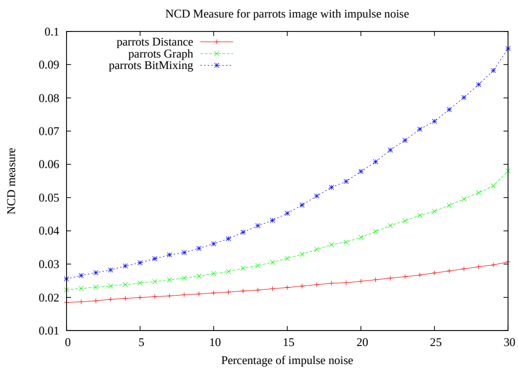
As for morphological filtering, the reference color $\mathbf{x}_{ref}$ used in the differentiation of the $\vee$ and the $\wedge$ can have some influence on median filtering. Figure 11 presents a study of the influence of the reference color on vector median filtering with a



(a) Parrots MSE



(b) Parrots MAE



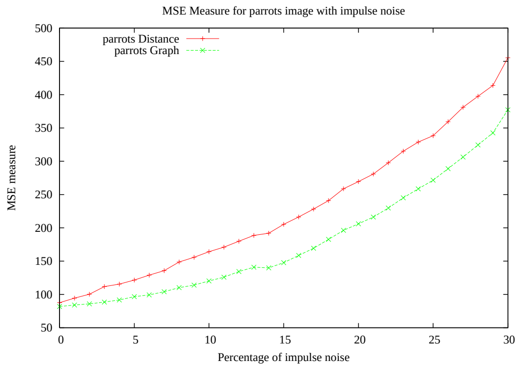
(c) Parrots NCD

Fig. 8. VMF Error Measures (MSE, MAE and NCD) with different orderings having the respective following curve colors: blue for bit mixing, green for graph and red for reduced.

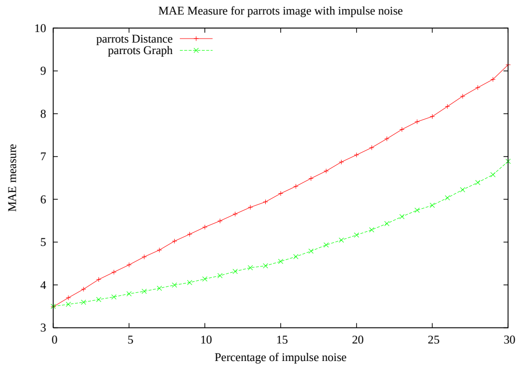
graph-based ordering and different reference colors among red, green, blue, cyan, yellow, purple, black and white. This influence is studied in the presence of impulse noise. As shown in figure 11, the reference color does not have a lot of influence on the results and black as color reference is sufficient in most of the cases. As noise increases the best reference color can change (cyan in this case). However, changing the reference color still does not make it possible to outperform the vector median filtering with reduced ordering based on distances.

6. Conclusion

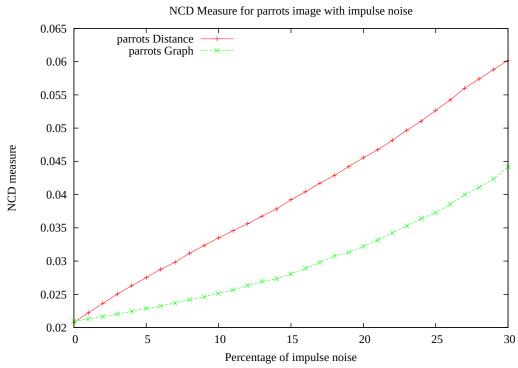
A new ordering of vectors based on the construction of a Hamiltonian path across the pixels of a filtering window has been proposed. A Hamiltonian path is equivalent to a space filling curve and we dynamically construct such a path on a given filter window. The proposed method is based on a two-step analysis of the color vectors



(a) Parrots MSE



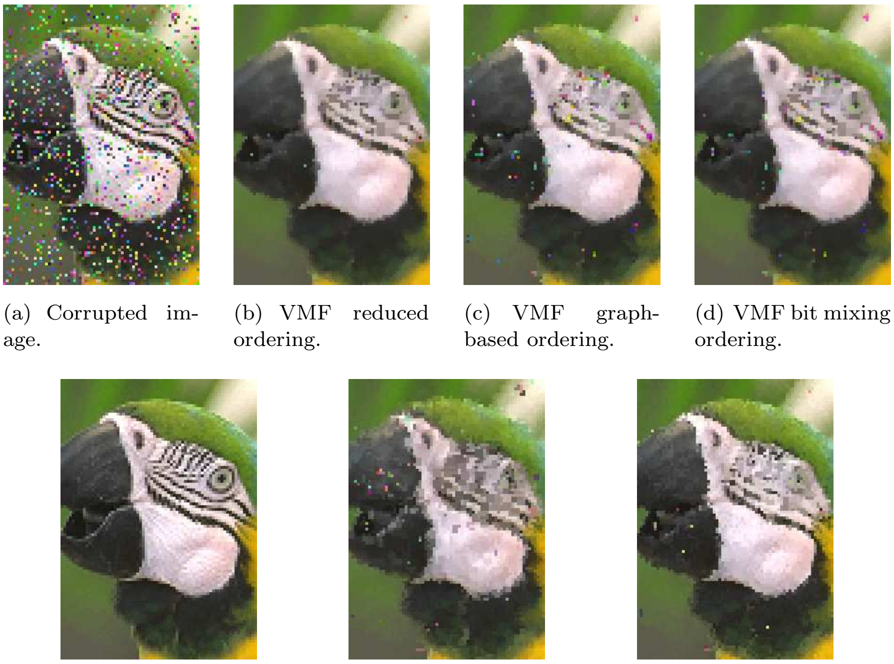
(b) Parrots MAE



(c) Parrots NCD

Fig. 9. DDF Error Measures (MSE, MAE and NCD) with different orderings.

within a filter window. The first step extracts the two extreme vectors of a set of color vectors. Starting from the minimum spanning tree of a complete graph defined on a filter window, an iterative selection of candidate vectors is performed. The latter is based on the degree of the vertices of the graph. From these two vectors, a Hamiltonian path is constructed on the complete graph representing the filter window. The proposed ordering has the advantage of being directly applicable to vectors of any dimensions and is suited for morphological or median filtering. However, the filtering abilities of a median filter based on the proposed graph-ordering are less efficient those of a classical vector median filter. The proposed ordering of vectors opens a new way of ordering color vectors via graph-theoretical algorithms. Future research will be about the reduction of the complexity of the proposed vector ordering because it is higher than that of usual algorithms.



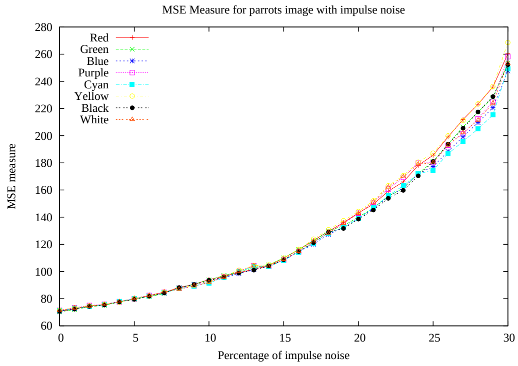
(a) Corrupted image. (b) VMF reduced ordering. (c) VMF graph-based ordering. (d) VMF bit mixing ordering.

(e) Original image. (f) DDF reduced ordering. (g) DDF graph-based ordering.

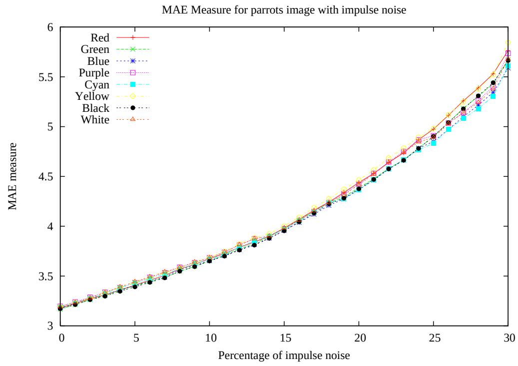
Fig. 10. 15% impulse noise filtered output with a VMF and a DDF and different vector ordering schemes.

References

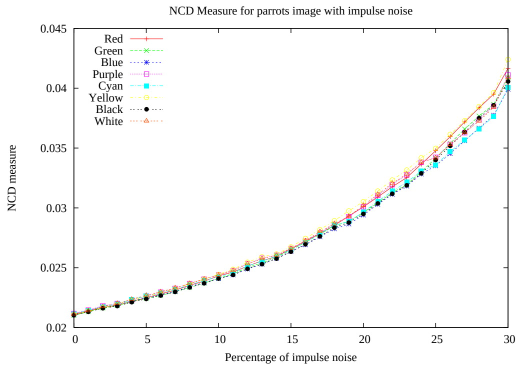
1. J. Angulo. Morphological color processing based on distances. Application to color denoising and enhancement by centre and contrast operators. In *Proc. of VIIP*, pages 314–319, 2005.
2. J. Angulo. Unified morphological color processing framework in a lum/sat/hue representation. In *Proc. of ISMM*, pages 387–396, 2005.
3. J. Angulo and J. Serra. Morphological coding of color images by vector connected filters. In *Proceedings of ISSPA*, volume 1, pages 69–72, 2003.
4. J. Astola, P. Haavisto, and Y. Neuvo. Vector median filters. *Proceedings of the IEEE*, 74(4):678–689, April 1990.
5. V. Barnett. The ordering of multivariate data. *Journal of the royal society of statistics*, A 139(3):318–355, 1976.
6. J. Chanussot and P. Lambert. Total ordering based on space filling curves for multivalued morphology. In *Proc. of ISMM*, pages 51–58, 1998.
7. M.L. Comer and E.J. Delp. Morphological operations for colour image processing. *Journal of electronic imaging*, 8(3):279–289, 1999.
8. R. Diestel. *Graph Theory*, volume 173. Springer-Verlag, 2005.
9. John K. Goutsias, Henk J. A. M. Heijmans, and Krishnamoorthy Sivakumar. Morphological operators for image sequences. *Computer Vision and Image Understanding*, 62(3):326–346, 1995.



(a) Parrots MSE



(b) Parrots MAE



(c) Parrots NCD

Fig. 11. Error Measures (MAE, MSE, NCD) for a VMF with a Graph-based ordering and different reference colors.

10. A. Hanbury. *Mathematical Morphology on the Unit Circle: with applications to hues and to oriented textures*. PhD thesis, Centre de Morphologie Mathematique, Ecole des Mines de Paris, 2002.
11. A. Hanbury and J. Serra. Mathematical morphology in the HLS colour space. In *Proc. of BMVC'2001*, volume 2, pages 451–460, 2001.
12. M. Iwanowski and J. Serra. Morphological interpolation and color images. In *Proceedings of the 10th International Conference on Image Analysis and Processing*, pages 50–, 1999.
13. D.G Karakos and P.E. Trahanias. Combining vector median and vector directional filters: the directional-distance filters. In *Proceedings of the International Conference on Image Processing*, pages 171–174, 1995.
14. M. Köppen, Ch. Nowack, and G. Rosel. Pareto-morphology for color image processing. In *The 11th Scandinavian Conference on Image Analysis*, volume 1, pages 195–202, 1999.
15. P. Lambert and J. Chanussot. Extending mathematical morphology to color image processing. In *First Int. Conf. on Color in Graphics and Image Processing*

(CGIP'2000), pages 158–163, 2000.

16. P. Lambert and L. Macaire. Filtering and segmentation: the specificity of color images. In *First Int. Conf. on Color in Graphics and Image Processing (CGIP'2000)*, pages 57–71, 2000.
17. O. Lezoray, C. Meurie, and A. Elmoataz. A graph approach to color mathematical morphology. In *Proc. of ISSPIT*, pages 856–861, 2005.
18. G. Louverdis, M.I. Vardavoulia, I. Andreadis, and Ph. Tsalides. A new approach to morphological color image processing. *Pattern recognition*, 35(8):1733–1741, 2002.
19. R. Lukac. Adaptive vector median filtering. *Pattern Recognition Letters*, 24(12):1889–1899, 2003.
20. R. Lukac and K. Plataniotis. *Color Image Processing: Methods and Applications*, volume 7. CRC Press, 2006.
21. R. Lukac and K.N. Plataniotis. A taxonomy of color image filtering and enhancement solutions. In Peter Hawkes, editor, *Advances in Imaging and Electron Physics*, volume 140, pages 187–264. Elsevier, 2006.
22. R. Lukac, K.N. Plataniotis, B. Smolka, and A.N. Venetsanopoulos. Generalized selection weighted vector filters. *EURASIP Journal on Applied Signal Processing*, 2004(12):1870–1885, 2004.
23. R. Lukac, B. Smolka, K. Martin, K.N. Plataniotis, and A.N. Venetsanopoulos. Vector filtering for color imaging. *IEEE Signal Processing Magazine, Special Issue on Color Image Processing*, 22(1):74–86, 2005.
24. R. Lukac, B. Smolka, K.N. Plataniotis, and A.N. Venetsanopoulos. Selection weighted vector directional filters. *Computer Vision and Image Understanding*, 94(1-3):140–167, 2004.
25. F. Ortiz, F. Torres, J. Angulo, and S. Puente. Comparative study of vectorial morphological operations in different color spaces. In *Proc. of Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision*, volume 4572, pages 259–268, 2001.
26. F. Ortiz, F. Torres, S. Puente, F. Candelas, and P. Gil. Use of the hue/saturation/intensity color spaces to the morphological. In *Proceedings de First International Conference on Color in Graphics and Image Processing*, pages 219–224, 2000.
27. R. A. Peters. Mathematical morphology for angle valued images. In *SPIE International Conference on Electronic Imaging*, 1997.
28. I. Pitas and P. Tsalides. Multivariate ordering in color image filtering. *IEEE Transactions on Circuits and Systems for Video Technology*, 1:247–259, 1991.
29. K. Plataniotis and A.N. Venetsanopoulos. *Color Image Processing and Applications.* Springer Verlag, 2000.
30. C.S. Regazzoni and A. Teschioni. A new approach to vector median filtering based on space filling curves. *IEEE Trans. on Image Processing*, 6:1025–1037, 1997.
31. C. Ronse. Why mathematical mophology needs complete lattices. *Signal Processing*, 21(2):129–154, 1990.
32. A. Soria-Frisch and M. Koppen. The fuzzy integral as similarity measure for a new color morphology. In *Proc. of CGIV*, pages 523–526, 2002.
33. J. Steele, L. Shepp, and W. Eddy. On the number of leaves of a euclidean minimal spanning tree. *J. Appl. Proba.*, 24:809–826, 1986.
34. Elena Stringa, Andrea Teschioni, and Carlo S. Regazzoni. A classical morphological approach to color image filtering based on space filling curves. In *Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP'99)*, pages 351–354, 1999.

35. H. Talbot, C. Evans, and R. Jones. Complete ordering and multivariate mathematical. In *Proceedings of the fourth international symposium on Mathematical morphology and its applications to image and signal processing*, pages 27–34, 1998.
36. Ch. Theoharatos, G. Economou, and S. Fotopoulos. Color edge detection using the minimal spanning tree. *Pattern recognition*, 38:603–606, 2005.
37. P.E. Trahanias, D. Karakos, and A.N. Venetsanopoulos. Directional processing of color images: Theory and experimental results. *IEEE Transactions on Image Processing*, 5:868–880, 1996.
38. P. Tsalides, M. I. Vardavoulia, and I. Andreadis. Vector ordering and morphological operations for colour image processing. *Pattern Anal. Appl*, 5(3):271–287, 2002.
39. M. I. Vardavoulia, I.Andreadis, and P. Tsalides. A new vector median filter for colour image processing. *Pattern Recognition Letters*, 22(6):1733–1741, 2001.
40. C. Vertan, M. Malciu, T. Zaharia, and V. Buzuloiu. A clustering approach to vector mathematical morphology. In *Proc. of the IEEE International Conference on Electronics, Circuits and Systems ICECS 96*, volume 1, pages 187–190, 1996.

Acknowledgements

This work was supported by the University of Caen Low Normandy under grants of FEDER and FNADT funds and by the ANR foundation under grant ANR-06-MDCA-008-01/FOGRIMMI

Photo and Bibliography

Olivier Lézoray received the M.S and Doctoral degrees, both in Computer Science, from University of Caen Basse-Normandie, Caen, France, respectively in 1996 and 2000. From 1999 to 2000, he was assistant professor with the Computer Science Department of the University of Caen. Since 2000, he has been an associate professor with the Cherbourg Institute of Technology (University of Caen) in the Communications, Networks and Services Department.
His research concerns color image segmentation and filtering (variational and morphological methods) and machine learning techniques for image mining (neural networks and support vector machines).

Cyril Meurie received his PhD from the University of Caen, France in 2005 in the field of computer science. From 2004 to 2006, he was assistant professor at the Computer Science Department of the University of Caen. Currently, he is a one-year postdoctoral researcher at the INRETS (French National Institute for Transport and Safety Research), Lille, France. His research focuses on image segmentation techniques for color images based on multi-scale approaches with application in transport.

Abderrahim Elmoataz is a full-time professor of computer Science at the Computer Science Departement of the University of Caen. His research focuses on PDE methods for image restoration and segmentation.