A New Approach to Vector Median Filtering Based on Space Filling Curves

Carlo S. Regazzoni, *Member, IEEE,* and Andrea Teschioni

***Abstract*— The availability of a wide set of multidimensional information sources in different application fields (e.g., color cameras, multispectral remote sensing imagery devices, etc.) is the basis for the interest of image processing research on extensions of scalar nonlinear filtering approaches to multidimensional data filtering.**

In this paper, a new approach to multidimensional median filtering is presented. The method is structured into two steps. Absolute sorting of the vectorial space based on Peano space filling curves is proposed as a preliminary step in order to map vectorial data onto an appropriate one-dimensional (1-D) space. Then, a scalar median filtering operation is applied. The main advantage of the proposed approach is the computational efficiency of the absolute sorting step, which makes the method globally faster than existing median filtering techniques. This is particularly important when dealing with a large amount of data (e.g., image sequences). Presented results also show that the filtering performances of the proposed approach are comparable with those of vector median filters presented in the literature.

I. Introduction

THE development of image processing methods for filtering and segmentation of vectorial information is becoming more and more important, thanks to the availability of a large amount of vectorial data. Noise filtering and enhancement of color images and of multispectral remote-sensing data are examples of applications where vectorial information must be processed.

Many sophisticated methods for scalar image processing are available that have been developed for grey-level image processing. A direct extension of such methods to the multivariate case can be performed by separately applying a scalar method to each component of a vectorial image. However, in many cases, this generalization exhibits several drawbacks, the most important of which lies in not exploiting interchannel correlation. A further consequence of this approach is that computational complexity increases linearly with the number of channels. Therefore, the study of new methods becomes necessary to realize efficient vectorial counterparts of scalar methods.

In this paper, an image filtering method is proposed. The proposed filter aims at extending median filters to the vectorial case by using a more flexible and more computationally efficient method.

Manuscript received April 18, 1996; revised February 28, 1997. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jan Allebach.

The authors are with the Department of Biophysical and Electronic Engineering (DIBE), University of Genoa, I-16145 Genoa, Italy (e-mail: carlo@dibe.unege.it).

Publisher Item Identifier S 1057-7149(97)04730-1.

In particular, the reduced vector median filter (RVMF) is introduced that has been derived directly from the concept of vector median filter (VMF). The VMF [1] is defined as the generalization of the scalar median filter [2] to the case of vector-valued signals.

The RVMF operates a vectorial-scalar transformation followed by scalar data ordering instead of directly realizing a median on the vectorial data, as for the VMF. The proposed transformation is based on the concept of space filling curves [3], [4].

In Section V, it is shown that our method exhibits a much lower computational complexity than other VMF implementations, and it is demonstrated that the RVMF performances are comparable to those of other VMF realizations.

The paper is organized as follows. Sections II and III provide a general description of the VMF and an introduction to the problem. Section IV focuses the attention on the chosen space filling curve and Section V shows the performances of the proposed algorithm. Section VI concludes the paper.

II. Vector Median Filters

The use of the median operator in image processing was introduced by Tukey [5]. The median filter performs a nonlinear filtering operation where a window moves over a signal, and, at each point, the median value of the data within the window is taken as the output.

The median of a scalar set $W = \{x_i : i = 1 \cdots n\}$ was defined as the value x_{MED} such that

$$x_{\mathrm{MED}} = \arg \min_{x_j \in W} \sum_i |x_i - x_j| \tag{1}$$

and it is known that x_{MED} can always be chosen as one of the x_i.

A statistical analysis of the median filter [2] revealed that it outperforms the moving average filter in the case of additive long-tailed noise, and that it is very suitable for the removal of impulsive noise. These facts make the median filter very attractive for digital image filtering applications.

Astola *et al.* [1] proposed an extension of the median operation to vector-valued signals, introducing some requirements for the resulting vector median.

The median value $\underline{x}_{\mathrm{MED}}$ of a vectorial set $W = \{\underline{x}_i : i = 1 \cdots n\}$, consisting of n p-variate samples, i.e., $\underline{x}_i = \{x_{i1}, x_{i2}, \cdots, x_{ip}\}$, is defined as

$$\underline{x}_{\mathrm{MED}} = \arg \min_{\underline{x}_j \in W} \sum_i ||\underline{x}_i - \underline{x}_j||. \tag{2}$$

The extended definition implies that the vectorial operator has properties similar to those of the median operation in the scalar case, and also implies that the vector median is reduced to the scalar median when the vector dimension is one. Such a definition of the VMF requires that the VMF output be one of the input vectors.

One can observe that this definition of the VMF heavily depends on the choice of an appropriate norm $\|\cdot\|$ for the vectorial values; the related distance is used to make a sorting of the quantities $\sum_i \|\underline{x}_i - \underline{x}_j\|$ corresponding to each vector within the set W.

The vector median can be computed by two different classes of algorithms, depending on the way in which the sorting is performed: absolute sorting or relative sorting.

The methods belonging to the first class use the same strategy for the median choice as in the scalar case: A scalar rank value is first associated with each vector on which the sorting is made.

For example, it is possible to consider the Euclidean distance of each vector $\underline{x}_j \in W$ from the origin of the p-dimensional space and then to make a sorting of the vectors within the set W according to this distance and to choose the median value from the distances, that is

$$\underline{x}_{\mathrm{MED}} = \mathrm{med}\,\{\|\underline{x}_j\|_p\} \tag{3}$$

where $\underline{x}_j \in W$ and $\|\cdot\|_p$ is an L_p norm.

A different choice (relative sorting) to implement the VMF may involve decomposing the minimum operation in (2) into the following two steps.

$$\begin{aligned} &1)\ \text{Find } \underline{x}_{\mathrm{PROT}} \equiv \{\text{Prototype vector within the set } W\} \\ &2)\ \underline{x}_{\mathrm{MED}} = \arg \min_{\underline{x}_j \in W} \|\underline{x}_{\mathrm{PROT}} - \underline{x}_j\|. \end{aligned} \tag{4}$$

The VMF implementations based on reduced ordering (r-ordering) [6] adopt this strategy in order to compute the VMF within the vectorial set W.

According to the r-ordering principle, multivariate ordering is reduced to a scalar one, where the scalar is a distance function of the multivariate samples to a central location (i.e., the prototype).

The VMF implementations based on this concept may differ in the choice of the prototype, which can be, for example, the mean of the vectors ($\underline{x}_{\mathrm{PROT}} = \frac{1}{n}\sum_{i=1}^{n} \underline{x}_i$) (r-ordering) or the marginal median (m-ordering [7]), which is the result of a scalar median operation independently performed along each component of the multivariate samples.

The presented methods usually choose the prototype vector using the modulus information of each vector. Further implementations of VMF based on relative sorting are presented in [8] and [9]. In such algorithms, the prototype vectors are chosen on the basis of the angular distance between the vectors instead of rank information like that used by the previously presented vectors.

The main advantage of absolute sorting methods is that it is possible to compute the VMF in only one step, so they exhibit a much higher computational efficiency than relative sorting methods.

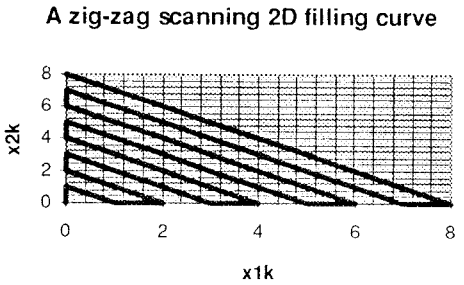


Fig. 1. Space filling curve used in image scanning algorithms.

A further consideration concerns the fact that absolute sorting can be easily extended to other kinds of filters, such as rank-order filters or morphological filters.

In this paper, we propose a new vector sorting approach. To this end, the concept of space filling curves is exploited. The proposed approach is of the absolute type, so the necessity for choosing a reference vector for sorting can be avoided.

The problem of mapping a multidimensional space into one-dimensional (1-D) space has long been considered an important one in the image processing literature. Space filling curves represent a possible solution to this problem. Space filling curves can be defined as a set of discrete curves that make it possible to cover all the points of a p-dimensional vectorial space. In particular, a space filling curve must pass through all the points of the space only once, and makes it possible to realize a mapping of a p-dimensional space into a scalar interval.

In Fig. 1 it is possible to observe quite a complicated two-dimensional (2-D) filling curve useful for an image scanning:

By means of a space filling curve, for example, a set of bidimensional elements may be reduced to a list of 1-D elements, representing the curvilinear abscissa of the 2-D points along the curve itself.

The space filling curves have been used in cryptography problems; Alexopoulos *et al.* [10] generated a family of scanning patterns for the protection of picture data in transmission. An image is scanned in a different way, as compared with the raster-scan technique: The result of this new scanning is the generation of a list of image pixels to which the inverse scanning can be applied in the detection phase in order to recover the original image. An image scanning method based on space filling curves has also been employed by Quweider and Salari [11] and by Kamata *et al.* [12] in order to obtain an algorithm for image compression and coding: This method allows one to fully exploit the correlation between adjacent pixels. Image coding algorithms based on space filling curves have been proven to be much better than raster methods.

III. Problem Definition

We used the concept of space filling curves as a starting point in order to derive a vector sorting algorithm. Let us consider, for example, the filling curve described above. As mentioned earlier, the curve makes it possible to cover all the points of a p-dimensional space continuously and once. It is then possible to associate with each point in the p-dimensional space a scalar value that is directly proportional to the length

of the curve necessary to reach the point itself starting from the origin of the coordinates. Vector sorting may be reduced to the simple sorting of the scalar values associated with each vector.

The problem is stated as follows: Let W be the set of vectors that have to be sorted. As previously said, the VMF can be defined as the vector $\underline{x}_{\mathrm{MED}} \in W$ that minimizes the cost function

$$J(\underline{x}_j) = \sum_i \|\underline{x}_i - \underline{x}_j\|. \tag{5}$$

That is

$$\underline{x}_{\mathrm{MED}} = \arg \min_{\underline{x}_j \in W} J(\underline{x}_j). \tag{6}$$

The proposed filter is the RVMF, and its output is the vector $\underline{x}_{\mathrm{MED}}$ that minimizes the cost function defined as

$$J(T(\underline{x}_j)) = \sum_i |T(\underline{x}_i) - T(\underline{x}_j)| \tag{7}$$

where $T : Z^p \rightarrow Z$ is a biunivocal transformation that allows a vectorial-scalar transformation, and in our case

$$\underline{x}_{\mathrm{MED}} = \arg \min_{\underline{x}_j \in W} J(T(\underline{x}_j)). \tag{8}$$

The only requirement to be met is that the function T be invertible, i.e., that T^{-1} exist.

The main advantage of the choice of such a cost function is that it is not necessary to define an appropriate vector norm $\|\cdot\|_p$ for the p-dimensional space, as the L_1 norm can be univocally chosen.

The algorithm by which we can find the vector $\underline{x}_{\mathrm{MED}}$ may be decomposed into three steps:

$$\begin{aligned} &1)\ \text{Compute } T(\underline{x}_i) \text{ for all } \underline{x}_i \in W \\ &2)\ T(\underline{x}^*) = \mathrm{med}\,\{T(\underline{x}_i)\} \\ &3)\ \underline{x}_{\mathrm{MED}} = T^{-1}(T(\underline{x}^*)). \end{aligned} \tag{9}$$

The first step is the application of the function $T(\cdot)$ to each vector belonging to the set W, followed by a scalar median operation performed on the set of transformed values $T(\underline{x}_i)$: The result $\underline{x}_{\mathrm{MED}}$ provided by the RVMF is eventually given by the antitransformed value of the median among the transformed values.

Before carrying out the minimum operation that leads to the vector median, we introduce into the proposed definition a vectorial-scalar transformation that allows us to simplify the solution of the problem itself. Through the use of filling curves, we also obtain a remarkable reduction in computational complexity, as compared with previous algorithms for vector sorting, and a considerable decrease in the processing time, thus obtaining comparable results, even though with a loss in terms of signal-to-noise ratio (SNR) performances.

IV. The Chosen Space Filling Curve

A space filling curve γ, associated with a 2-D $M \times N$ lattice S allows the association of a scalar value with a p-dimensional vector in a discrete vectorial space K as

$$\begin{aligned} &1)\ \gamma : Z \rightarrow K, \quad K \subset Z^p \\ &2)\ \gamma(t_k) = \underline{x}_k(t_k) = \{x_{1k}(t_k), \cdots, x_{pk}(t_k)\} \\ &3)\ t_k \in Z, \quad \underline{x}_k \in K. \end{aligned} \tag{10}$$

For each $\underline{x}_k$, the *arc length* $\Gamma(t_k)$ is defined as

$$\Gamma(t_k) = \int_0^{t_k} \|\dot{\gamma}(\tau)\|\, d\tau = \int_0^{t_k} \sqrt{\sum_{i=1}^{p} [\dot{x}_{ik}(\tau)]^2}\, d\tau \tag{11}$$

where $\dot{\gamma}$ exists, whereas the parameter t_k represents the *curvilinear abscissa* of the curve.

A filling curve makes it possible to cover, as the parameter t_k varies, all the points of the discrete vectorial space K, so that each point is crossed only once, i.e.,

$$\begin{aligned} &\forall \underline{x}_k \in K \exists t_k : \gamma(t_k) = \underline{x}_k(t_k) \\ &\text{If } t_k, t_l \in Z \text{ and } t_k \neq t_l \text{ then } \gamma(t_k) \neq \gamma(t_l). \end{aligned} \tag{12}$$

In accordance with (12), a filling curve substantially makes a scanning operation of the K space and it generates a list of vectors $\underline{x}_k \in I$ in which there is no repetition of the same element $\underline{x}_k$.

An important consequence of the definition of filling curve is that the curve is invertible, i.e.,

$$\begin{aligned} &\text{If } \gamma(t_k) = \underline{x}_k \text{ then} \\ &\exists \gamma^{-1} : K \rightarrow Z : \gamma^{-1}(\underline{x}_k) = t_k. \end{aligned} \tag{13}$$

So far, the characteristics of a general space filling curve have been introduced. Let us now examine the required properties of the specific class of filling curves used in this paper to define the RVMF.

The curve we are searching for should reflect a "natural" ordering of vectors in the K space. In particular, it should be regular, and it should avoid irregular jumps similar to the ones in the curve of Fig. 1.

In fact, a filling curve like the one of Fig. 1 presents two main disadvantages, as follows, due to the fact that jumps between close vectors in the K space are not uniform:

- There is a "suboptimal" mapping of closeness properties in the K space onto the 2-D filling space (i.e., ordering does not follow a natural, continuous rule, but it is subject to "jumps").
- Computation of the arc length could be complex.

The first point expresses a local constraint on the absolute ordering we are searching for. In particular, we can define an optimal mapping in the following way.

Let us define $N_K(\underline{x}) = \{\underline{y} : \underline{y} \in K, \|\underline{y}, \underline{x}\| < d\} \backslash \{\underline{x}\}$ to be a neighborhood set of the point $\underline{x}$ in the K space and $N_Z(t_k) = \{t_m : |t_m - t_k|^2 < d'\} \backslash \{t_k\}$ be a neighborhood set of t_k points in the Z space.

Then we say that a space filling curve is optimal if all the points belonging to the neighborhood set of t_k in the Z space are close to $\underline{x}$ (i.e., $\underline{x}$ such that $\gamma(\underline{x}) = t_k$) in the K space, too.

In a formal way it can be expressed as

$$\begin{aligned} &\mathrm{card}\{t_i : t_i \in N_Z(t_k), \gamma(\underline{x}) = t_k, \exists \gamma(\underline{y}) = t_i, y \in N_k(\underline{x})\} \\ &\quad = \mathrm{card} N_Z(t_k) \end{aligned} \tag{14}$$

For example, if $d' = 1$, then card $N_Z(t_k) = 2$, so that it is possible to say that the optimality constraint implies that there exist two points $\underline{y}_1$ and $\underline{y}_2 \in N_K(\underline{x})$ that are close to $\underline{x}$ in the K space and which map onto the two elements of $N_Z(t_k)$. This clearly does not hold in the curve of Fig. 1.

There does not exist a unique space filling curve that satisfies the above requirements, so that it is necessary to introduce some additional criteria. To this end, we note that the desired curve should be able to generate a sorting of the K space points, which maintains, at best, modulus and angular information.

In this paper, we define a strategy to design a curve where a greater weight is given to the first requirement (i.e., rank preservation according to modulus information).

Let us define the $\|\cdot\|_{\max}$ of $\underline{x}_k$ as

$$\|\underline{x}_k\|_{\max} = \max\{x_{1k}, \cdots, x_{pk}\} \tag{15}$$

According to (15), the following property has to be valid.

Given two vectors $\underline{x}_k, \underline{x}_l \in I$

$$\begin{aligned} &\gamma(t_k) = \underline{x}_k, \gamma(t_l) = \underline{x}_l, \\ &\text{If } \|\underline{x}_k\|_{\max} > \|\underline{x}_l\|_{\max} \text{ then } t_k > t_l. \end{aligned} \tag{16}$$

In such a way, a higher value of the parameter t_k will be associated with vectors having a larger $\|\cdot\|_{\max}$ distance from the origin, and vice versa. The resulting curve can be imagined as consisting of successive layers, increasingly ordered according to the $\|\cdot\|_{\max}$ value.

Within each layer, the curve is chosen in such a way as to preserve angular information at the best. Among the set of space filling curves satisfying (14) and (16), a particular filling curve has been chosen, according to the latter observation.

The case of a bidimensional space $(p = 2)$ will be first illustrated for two reasons. The first one is that some concepts and demonstrations are easier to be understood in the 2-D case, and they can be extended to the three–dimensional (3-D) case with a direct generalization.

The second reason is that the 2-D case can be very interesting by itself because it has several applications (e.g., displacement field motion field filtering).

A. The 2-D Case

In the 2-D case, the filling curve is reduced to a function γ which associates with a scalar value a vector in the 2-D space, that is

$$\begin{aligned} &\gamma : Z \to K_2, \quad K_2 \subset Z^2, \quad K_2 = [0, X_{1\max}] \times [0, X_{2\max}] \\ &\gamma(t_k) = (x_{1k}(t_k), x_{2_k}(t_k)). \end{aligned} \tag{17}$$

The graphic representation of the chosen filling curve in the 2-D space (see also [13]) is the following.

The dependence of the components of each vector on the parameter t_k is expressed in a graphic way by these relations: The direct relations joining the components x_{1k} and x_{2k} of γ to the parameter t_k are

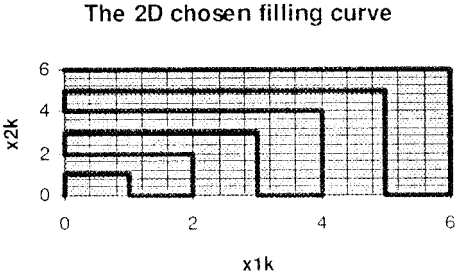


Fig. 2. Graphic representation of the chosen 2-D filling curve γ.

$$\begin{aligned} x_{1k}(t_k) &= \sum_{p=0}^{\left\lfloor \frac{X_{1\max}-2}{2} \right\rfloor} h_{2p+1}\left(t_k - (2p+1)^2\right) \quad t_k > 0 \\ x_{2k}(t_k) &= \sum_{p=0}^{\left\lfloor \frac{X_{2\max}-1}{2} \right\rfloor} h_{2p}\left(t_k - (2p)^2\right) \quad t_k > 0. \end{aligned} \tag{18}$$

The decomposition into function series of the two vector components reveals that they can be interpreted as a superposition of the same basis function $h_p(t)$ translated along the t_k axis.

Let us define the basis functions $h_0(t)$ and $h_1(t)$ as

$$h_0(t) = \begin{cases} t & t \in [0,1) \\ 1 & t \in [1,2) \\ -t+3 & t \in [2,3) \\ 0 & [3,4] \end{cases} \tag{19a}$$

$$h_1(t) = \begin{cases} t & t \in [0,1) \\ 1 & t \in [1,2) \\ t-1 & t \in [2,3) \\ 2 & t \in [3,5) \\ -t+7 & t \in [5,7) \\ 0 & t \in [7,8). \end{cases} \tag{19b}$$

Fig. 4(a) and (b) illustrates in a graphic way the behavior of $h_0(t)$ and $h_1(t)$.

The shape of the generic p-order basis function $h_p(t)$ is defined starting from the first order function $h_1(t)$ in the following way ($h_0(t)$ can be interpreted as an offset term depending on the chosen starting point of the curve):

$$h_p(t) = \begin{cases} ph_1(t/p) & t \in [0, 2p) \\ t-p & t \in [2p, 2p+1) \\ \frac{p+1}{2} h_1\left(\frac{2}{p+1}t - 2\frac{2p+1}{p+1} + 3\right) & t \in [2p+1, 4p+3) \\ 0 & t \in [4p+3, 4p+4). \end{cases} \tag{19c}$$

Using (19c) it is possible to observe that the shape of the basis function $h_p(t)$ is periodically repeated in (18) scaled up and dilated along the t_k axis.

From (18) and (19c) it is possible to demonstrate that the regularity of the reiterations of the basis functions strongly depends on the parameter t_k.

A further interesting property of the proposed filling curve γ is

$$\|\dot{\gamma}(t)\| = \sqrt{\dot{x}_{1k}^2(t) + \dot{x}_{2k}^2(t)} = 1 \quad \forall t \in Z \tag{20}$$

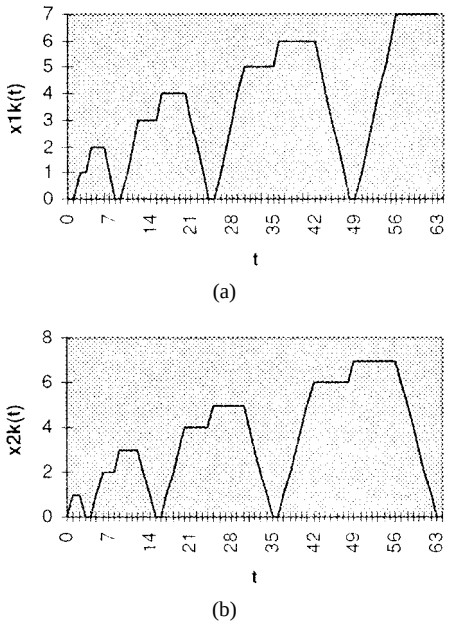


Fig. 3 (a)–(b) 2-D components of the chosen filling curve γ.

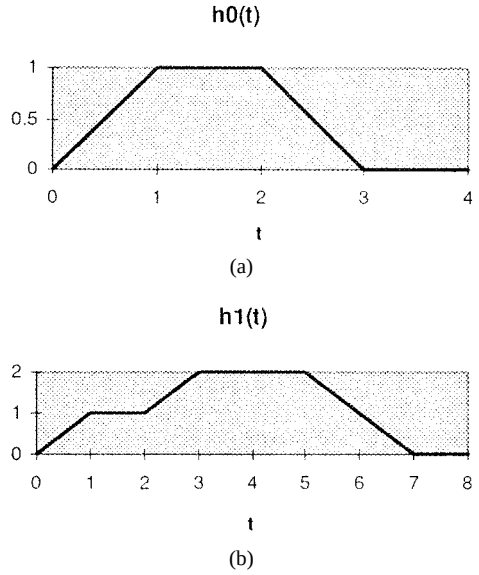


Fig. 4. (a)–(b) Graphic representation of the 2-D basis functions.

where we define $\dot{x}_{ik}(t) = \dot{x}_{ik}(t_+)$ and that

$$\dot{x}_{1k}(t)\dot{x}_{2k}(t) = 0 \quad \forall t \in Z. \tag{21}$$

In the case of the chosen filling curve, the arc length of the curve γ is coincident with its curvilinear abscissa. Fig. 5 illustrates in a graphic way such properties.

It is now necessary to demonstrate that (14) and (16) are satisfied in the case of the chosen space filling curve. The chosen filling curve performs an optimal mapping for a $N_Z(t_k)$ neighborhood system with $d' = 1$, $d = 1$, so that card $N_Z(t_k) = 2$.

Let us see how, starting from a chosen vector $\underline{x}_k$, it is possible to calculate the corresponding value t_k of the curvilinear abscissa.

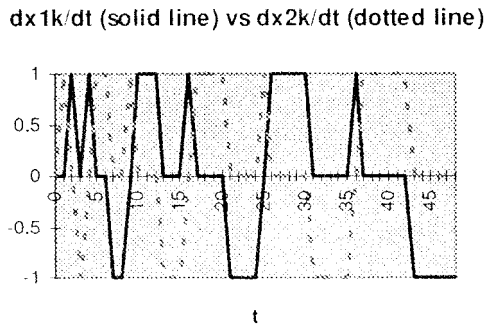


Fig. 5. Graphic representation of the derivatives of $x_{1k}(t)$ and $x_{2k}(t)$.

If $||\underline{\bar{x}}_k||_{\max} = \bar{x}_{1k} = A$, then, by definition of $h_p(t)$ and by construction

$$\exists l : h_l(\bar{t}) = A \quad \text{and} \quad \dot{x}_{1k}(\bar{t}) = 0 \quad \bar{t} \in Z. \tag{22}$$

The l value could be even or odd; it means that, if $l \in Z$ and $r \in Z$

$$(l = 2r + 1) \quad \text{or} \quad (l = 2r). \tag{23}$$

Let us suppose, without a loss of generality, that $(l = 2r+1)$; if we define

$$\bar{I} = \{l^2 + 2l + 1, l^2 + 3l + 2\} \tag{24}$$

we will have

$$A = \max_{t \in \bar{I}} h_l(t - l^2). \tag{25}$$

By construction and by definition of $h_p(t)$

$$\dot{x}_{1k}(t) \underset{t \in \bar{I}}{=} \dot{h}_l(t - l^2) \underset{t \in \bar{I}}{=} 0. \tag{26}$$

According to (20) and (26) it must be $\dot{y}(t)_{t \in \bar{I}} = \pm 1$ and $y(t)_{t \in \bar{I}} \in (0, A)$ and the following statement will hold:

$$\exists! \, \bar{t} \in \bar{I} : x_{2k}(\bar{t}) = \bar{x}_{2k}. \tag{27}$$

The obtained functions allow us to associate with each scalar value t_k a point in the 2-D space (x_{1k}, x_{2k}); conversely, it is possible to transform a point in the 2-D space into the corresponding scalar value t_k.

There is quite a complicated analytic relation that binds, in this case, the parameter t_k to the components x_{1k} and x_{2k} of γ: **div** is the integer division, **mod** is the remainder of the integer division, and $A = \max(x_{1k}, x_{2k})$, as follows:

$$\begin{aligned} t_k &= \gamma^{-1}(x_{1k}, x_{2k}) \\ &= \{\text{even}(x_{1k}, x_{2k}) + \text{odd}(x_{1k}, x_{2k}) \\ &\quad + \text{hist}(x_{1k}, x_{2k})\} * \left(\frac{A \,\text{div}\, 2}{\frac{A}{2} - A \,\text{mod}\, 2} \right) \\ &\quad + \max\_1(x_{1k}, x_{2k}) * \left(1 - \frac{A \,\text{div}\, 2}{\frac{A}{2} - A \,\text{mod}\, 2} \right). \end{aligned} \tag{28}$$

Expression (28) can be decomposed into four main terms, i.e., *even*, *odd*, *hist*, and Max_1: In Appendix A the relations between these terms and (18) and (19) definitions and the

Algorithm for calculation of the arc length of the 2D filling curve

```
begin
compute A = max(x_1k , x_2k)
if (A=0) then tkap = 0 endif
if (A=1) then tkap = max_1(x_1k , x_2k)
        else
        if (A even) then
                if (x_1k = A) then p_1 = x_2k endif
                if (x_2k = A) and (x_1k < A) then p_1 = 2*A-x_1k endif
        endif
        if (A odd) then
                if (x_2k = A) then p_1 = x_1k endif
                if (x_1k = A) and (x_2k < A) then p_1 = 2*A-x_2k endif
        endif
        tkap = p_1 + A^2
endif
return tkap
```

Fig. 6. C-like algorithm for the computation of γ^{-1} in 2-D case.

demonstration of validity of (16) for the chosen filling curve are presented.

Consequently, the ranking of vectors can be performed according to the *arc length* of the filling curve at different 2-D space points.

The C-like algorithm for the computation of the arc-length is described in Fig. 6.

The algorithm is simple and computationally efficient.

B. The 3-D Case

The generic expression of the 3-D space filling curve is expressed by

$$\begin{aligned} &\gamma : Z \rightarrow K_3, \quad K_3 \subset Z^3, \\ &K_3 = [0, X_{1\max}] \times [0, X_{2\max}] \times [0, X_{3\max}] \\ &\gamma(t_k) = (x_{1k}(t_k), x_{2k}(t_k), x_{3k}(t_k)). \end{aligned} \tag{29}$$

To design a space filling curve able to be used for color image processing it is necessary to extend results obtained in the 2-D case to the 3-D case.

The extension of the 2-D filling curve to the 3-D space is performed in order to preserve, as much as possible, the characteristics of the described curve for the 2-D case: in particular, in a similar way with respect to the 2-D case, the 3-D filling curve can be imagined as an expansion of successive increasing layers, ordered according to the $\|\cdot\|_{\max}$ value of each 3-D vector.

A possible strategy is to impose that the 3-D filling curve crosses all points at the same $\|\cdot\|_{\max}$ value in a continuous way, e.g., by covering in a ordered way the three sides of a cube.

In this way, it is possible to think that the behavior of the 3-D filling curve on a single side of the cube having $\|\cdot\|_{\max} = A$ will reflect the behavior that the 2-D filling curve follows in the 2-D space.

Moreover, it was shown in the 2-D case that it is possible to find a relation which allows to associate a scalar value t_k, representing the arc length of the filling curve, with a vector $\underline{x}_k \in Z^2$.

In Appendix A it is also shown that such a relation is composed by a term representing the history of the curve up to the considered layer and by an "updating" term, which gives the shift of the vector $\underline{x}_k$ within the layer itself and that the *hist* term increases in a quadratic way according to $\|\underline{x}_k\|_{\max}$.

This is mainly due to the fact that, as it can be easily seen from Fig. 2, the arc length of the curve from the origin to one of the points $(0, A)$ or $(A, 0)$ is equal to the number of points of the square of side A, that is A^2.

From Fig. 2 it is also possible to note that the behavior of the curve in the 2-D space can be considered as an oscillatory path from one coordinate axis to the other one.

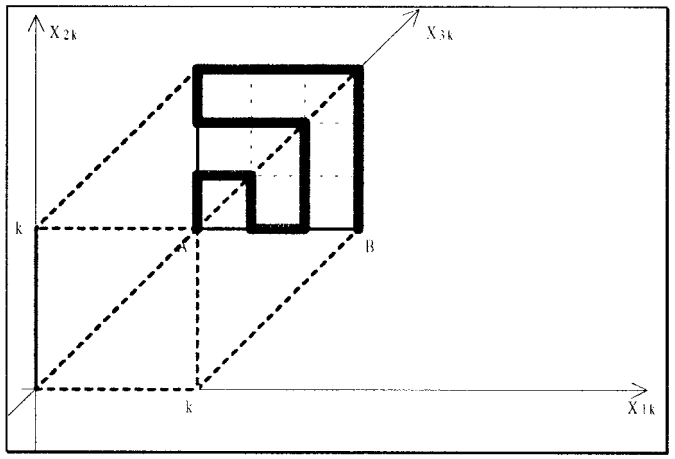
This behavior is reflected into the basis function $h_p(t)$ as a scaled-up mirroring of the semitrapezoidal shape corresponding to the path itself.

A curve with a similar behavior in the 3-D case is shown in Fig. 7.

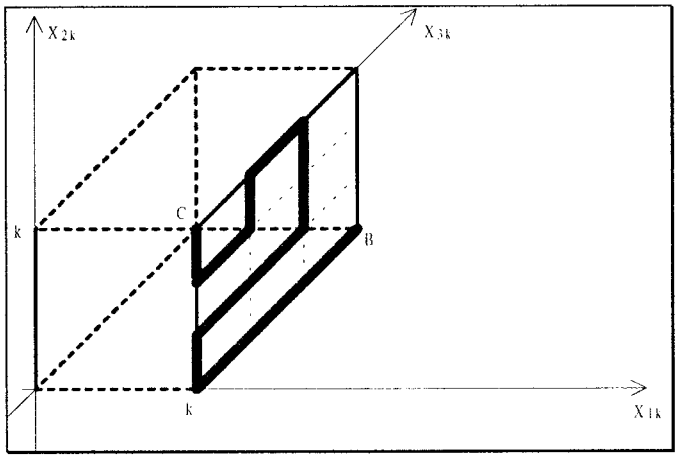
On this basis, we can define the curve of Fig. 7 in an analytic way as

$$\begin{aligned} x_{1k}(t_k) &= \sum_{p=0}^{\left\lfloor \frac{X_{1\max}-3}{3} \right\rfloor} h^{3D}_{3p+2}\left(t_k - (3p+2)^3\right) \quad t_k > 0 \\ x_{2k}(t_k) &= \sum_{p=0}^{\left\lfloor \frac{X_{2\max}-2}{3} \right\rfloor} h^{3D}_{3p+1}\left(t_k - (3p+1)^3\right) \quad t_k > 0 \\ x_{1k}(t_k) &= \sum_{p=0}^{\left\lfloor \frac{X_{3\max}-1}{3} \right\rfloor} h^{3D}_{3p}\left(t_k - (3p)^3\right) \quad t_k > 0 \end{aligned} \tag{30}$$

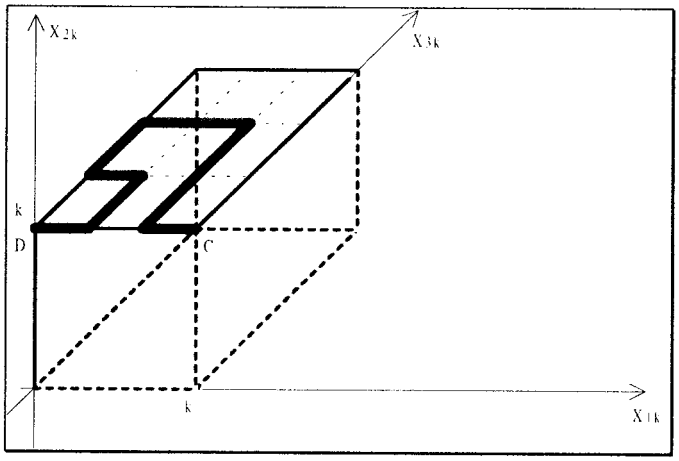
where a basis function $h^{3D}_p(t)$ is used.



(a)



(b)

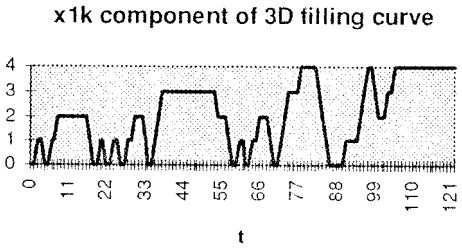


(c)

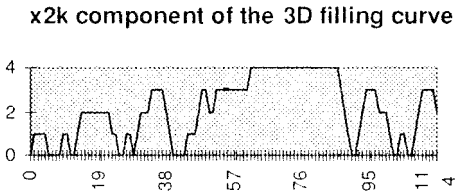
Fig. 7. Graphic representation of the chosen 3-D filling curve γ: the curve starts in A in (a) and ends in B; the curve starts in B in (b) and ends in C; the curve starts in C in (c) and ends in D.

The parametric behaviors of the components of the 3-D vector scanned by the curve are represented in a graphic way as shown in Fig. 8(a)–(c).

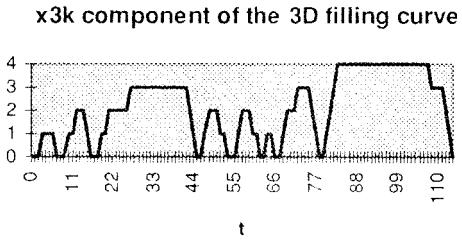
These relations can be considered as a generalization of expression (18).



(a)



(b)



(c)

Fig. 8. (a)–(c) 3-D components of the chosen filling curve γ.

A regularity is also present in the 3-D components, as in the 2-D case: This corresponds to a well-defined path followed by the 3-D curve on the different sides of the cube.

The possible behaviors of the $x_k(t)$ coordinates expressed by the basis function are a periodic repetition, scaling up, mirroring and dilation along the t_k axis.

In particular, we can observe that the coordinate $x_{1k}(t)$ remains constant $(\mathbf{C})$ when the side of the cube corresponding to the plane $x_{2k}x_{3k}$ is crossed.

To better understand the behavior of $h_p^{3D}(t)$, it is possible to fix a certain layer p of the 3-D cube and to examine the dependence of each component of the 3-D vector on the t_k parameter within the layer p.

If the curve of Fig. 8 is chosen, the following graphic expression is obtained, for example, for $x_{1k}(t_k)$ at the layer $p = 3$.

In general, each 3-D component can be expressed as a series function depending on the basis function $h_p(t)$: This is consistent with the fact that, on a single 3-D layer, the behavior of the curve on each side of the cube is the same followed by the 2-D curve in order to cover all the points of the 2-D space.

Along the same face, the other coordinates correspond to a scaled-up and truncated version of either a normal $(\mathbf{N})$ or a mirrored $(\mathbf{M})$ version of the basis shape [see Fig. 4(b)] depending on the direction by which the curve crosses the plane (see Fig. 10). In particular, if we consider the path for $x_{1k}(t)$ shown in Fig. 9(a), the three versions $\mathbf{C}$, $\mathbf{M}$, and $\mathbf{N}$ can be individuated:

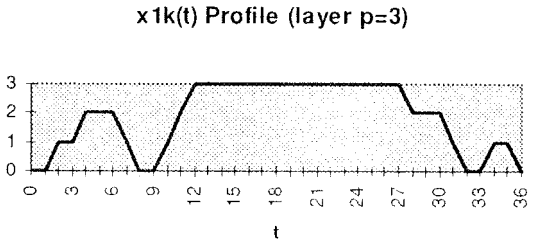
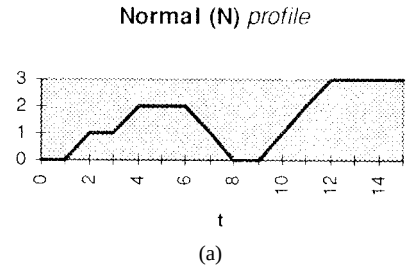
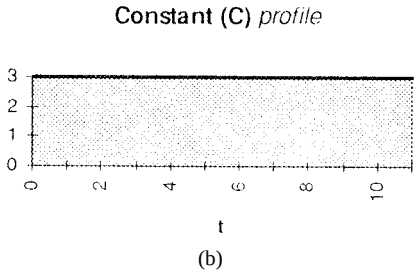


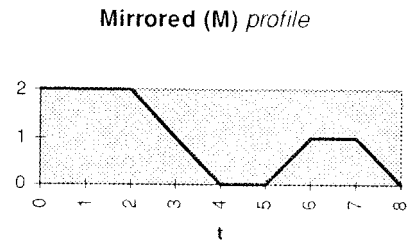
Fig. 9. Graphic representation of the profile of $x_{1k}(t)$ at layer $p = 3$.



(a)



(b)



(c)

Fig. 10. $\mathbf{C}$, $\mathbf{M}$, and $\mathbf{N}$ profiles individuated on the profile of Fig. 9.

On the basis of this description it can be shown that a coordinate crossed by the filling curve periodically describes a well-defined cycle in which it can be found in one of three statuses (i.e., $\mathbf{C}$, $\mathbf{N}$ or $\mathbf{M}$).

It is also possible to observe from Fig. 7 that the same behavior of the curve in the 3-D space is repeated after a number of layers equal to the dimension of the input vector (i.e., three).

This justifies the periodicity by which the different coordinates are expressed in (30).

It is also possible to demonstrate that, also in the 3-D case, properties (20) and (21) can be generalized to

$$||\dot{\gamma}(t)|| = \sqrt{\dot{x}_{1k}^2(t) + \dot{x}_{2k}^2(t) + \dot{x}_{3k}^2(t)} = 1 \quad \forall t \in Z \tag{31}$$

$$\dot{x}_{1k}(t)\dot{x}_{2k}(t) = 0 \quad \forall t \in Z \tag{32a}$$

$$\dot{x}_{1k}(t)\dot{x}_{3k}(t) = 0 \quad \forall t \in Z \tag{32b}$$

$$\dot{x}_{2k}(t)\dot{x}_{3k}(t) = 0 \quad \forall t \in Z. \tag{32c}$$

Also in this case, the arc length of the curve is coincident with the curvilinear abscissa.

The analytic relation that gives the arc-length in the 3-D case is

$$\begin{aligned} t_k &= \gamma^{-1}(x_{1k}, x_{2k}, x_{3k}) \\ &= \{\mathrm{mod\_0}(x_{1k}, x_{2k}, x_{3k}) + \mathrm{mod\_1}(x_{1k}, x_{2k}, x_{3k}) \\ &\quad + \mathrm{mod\_2}(x_{1k}, x_{2k}, x_{3k}) + hist(x_{1k}, x_{2k}, x_{3k})\} \\ &\quad * \left(\frac{A \,\mathbf{div}\, 2}{\frac{A}{2} - A \,\mathbf{mod}\, 2} \right) + \max\_1(x_{1k}, x_{2k}, x_{3k}) \\ &\quad * \left(1 - \frac{A \,\mathbf{div}\, 2}{\frac{A}{2} - A \,\mathbf{mod}\, 2} \right). \end{aligned} \tag{33}$$

Also in this case, **div** is the integer division, **mod** is its remainder, and $A = \max(x_{1k}, x_{2k}, x_{3k})$.

The $\mathrm{mod\_0}$ is nonzero only if $A \,\mathbf{mod}\, 3$ is equal to zero, and so on for $\mathrm{mod\_1}$ and $\mathrm{mod\_2}$.

The *hist* term contains the previous history and

$$hist(x_{1k}, x_{2k}, x_{3k}) = A^3. \tag{34}$$

$\mathrm{Max\_1}$ is nonzero if $A = 1$ and $t_k = 0$ iff x_{1k} and x_{2k} and x_{3k} are equal to zero.

We can see that (31) is of the same kind of (28), even if, in the 3-D case, the *hist* term obeys to a cubic law because of the fact that in this case all the points of a cube will to be covered by the filling curve. From Fig. 7, one can easily see that the arc length of the curve from the origin to one of the points $(0, 0, A)$, $(0, A, 0)$ or $(A, 0, 0)$ is equal to the number of points of the cube of side A, that is, l^3.

In the 3-D case, in order to calculate the arc length of the chosen filling curves, three operations will be necessary: A maximum computing in order to find which $\mathrm{mod\_}i(\cdot)$ has to be selected, another maximum computing, made only on two numbers (i.e., the two coordinates that do not correspond to the maximum previously calculated), which corresponds to the maximum calculation made in the case of the 2-D filling curve, and finally a shift calculation which, as in the 2-D case, normally consists in an addition operation.

We could expect that in the case of $K \in Z^p$ the relation binding the parameter t_k to the p-D components of a filling curve γ appropriately designed would be composed by a hist term A^p (e.g., A is the side of the layer of the p-D hypercube) and by a term which gives the shift of the considered vector within the layer with side A.

More in general, in order to calculate the arc length of a p-dimensional filling curve designed by extending the 2-D and 3-D presented curves, it will be necessary to progressively transform a p-dimensional vectorial space by computing $(p-1)$ maxima within sets whose dimension is progressively decreased: At the $(p-1)$th calculation, the vectorial space is reduced to a scalar one and the "updating" term to be added to hist can be calculated.

In a more complicated way than in the 2-D case, it can be demonstrated that such a choice of the filling curve will satisfy the properties (14) and (16). In particular, in order to demonstrate the validity of (16), we can refer to the demonstration of Appendix A, because if we choose the particular side of

the 3-D on which we are, we come back, as we already said, to the 2-D situation, and then the demonstration of Appendix A is valid.

The C-like algorithm to compute the arc-length in the 3-D case is shown in Fig. 11.

C. The 3-D RVMF

Immediately successive to the choice of the filling curve is the introduction of the transformation, which is defined for a vector $\underline{x}_k$ and for the chosen 3-D filling curve as

$$T(\underline{x}_k) = \gamma^{-1}(\underline{x}_k) = t_k. \tag{35}$$

After having computed, for each vector $\underline{x}_k$ the $T(\underline{x}_k)$ value, it is possible to rank the vectors $\underline{x}_i$ within a mask and to compute the scalar median as in [2].

The RVMF output is given by

$$\underline{x}_{\mathrm{MED}} = \gamma(T(\underline{x}^*)). \tag{36}$$

where $T(\underline{x}^*) = \mathrm{med}(T(\underline{x}_i))$.

V. Results

In this section, an application of the proposed method to color image filtering is presented.

RVMF filter performances are evaluated, and RVMF is compared with

1) *marginal median* [7] (i.e., ordering performed along each component of the multivariate samples);
2) *r-ordering about the mean* [6] (i.e., ordering of the multivariate samples according to their distances to a preselected central location, in such case the mean of the multivariate samples);
3) *C (Conditional)-ordering* [6] (i.e., ordering of only one of the components—other components are simply listed according to the position of the ranked component);
4) *VMF with Square Euclidean Distance* [6] (i.e., ordering performed according to the square modulus of each vector);

in terms of signal-to-noise ratio (SNR). The test image selected for the comparison is the color version of Lena. The test image has been contaminated using Gaussian and "salt and pepper" impulsive noise source models: A correlation factor $\rho = 0.5$ between the components of the vectorial noise is used in the experiments.

The SNR has been used as quantitative measure for evaluation purposes. It is computed as

$$\mathrm{SNR} = 10\log_{10} = \frac{\sum_{k=0}^{N1}\sum_{l=0}^{N2}\left(\sum_{i=0}^{p} y_i^2(k,l)\right)}{\sum_{k=0}^{N1}\sum_{l=0}^{N2}\left(\sum_{i=0}^{p}(y_i(k,l)-\hat{y}_i(k,l))^2\right)} \tag{37}$$

where $N1$ and $N2$ are the image dimensions, $\underline{y}(k,l)$ and $\underline{\hat{y}}(k,l)$ and denote the original image vector and the estimation at pixel (k,l), respectively.

Table II summarizes the results obtained for the test image Lena for a filter window 3×3: The noise corruption types are illustrated in Table I.

TABLE I
Noise Distributions

Number	Noise model
1	Gaussian (σ=20)
2	Gaussian (σ=10) impulsive (1%)
3	Impulsive (2%)
4	Impulsive (4%)
5	Impulsive (6%)
6	Impulsive (10%)

One can observe that the filter produces comparable results with the r-ordering implementation, which is currently considered the filter that gives better performances, in the case of impulsive noise; as expected, the performances are a few worse for Gaussian noise, since the proposed transformation is based on a nonlinear distance.

Moreover, r-ordering performs a relative sorting, which usually produces better results than absolute sorting; a comparison with methods based on an absolute sorting has been also made.

For example, RVMF gives almost the same performances in terms of SNR as the marginal median, while performances of RVMF are usually better than c-ordering and VMF ones.

A comparison also has been made between RVMF and marginal median in terms of computational complexity; marginal median has been chosen because of its simplicity and because of the fact that it provides optimal computational performances among absolute sorting-based methods.

For each pixel of a D-dimensional $N \times N$ image a window with M points is considered. Marginal median requires $D \times M$ sorting operations: Since each sorting requires, as an average term, $L \times \log(L)$ comparisons [16] (L being the cardinality of the set to be sorted), then globally for the whole image, the marginal median will require $N \times N \times (D \times M \times \log(M))$ comparisons).

If a Sun SparcStation 20 is chosen, for each comparison one clock cycle is required [15]. Then marginal median operation will require $N \times N \times (D \times M \times \log(M))$ clock cycles.

For what concerns RVMF, it has been shown that the following operations are necessary for an $N \times N$ image:

1) $N \times N \times \{(D-1)$ maximum calculations $+1$ one addition$\}$, as an average;
2) $N \times N \times \{1$ scalar median operation$\}$.

One also has to consider that, for each maximum calculation the set of elements to be sorted is decreased each time of one element itself. The Sun SparcStation 20 workstation requires one clock cycle also for the addition, so the global number of clock cycles required by RVMF are

$$N \times N \times \left\{\sum_{i=2}^{D} i * \log i + 1 + M \log M\right\}. \tag{38}$$

Fig. 12 shows the behavior of the computational complexity (measured as clock cycles necessary for each pixel of an $N \times N$ image) of the two methods, by using as independent variable the D dimension of the space and after having fixed a mask dimension $M = 9$: One can see that the complexity is much lower in the case of the RVMF, even if there is the possibility

TABLE II
SNR (dB) FOR THE LENA IMAGE, WINDOW 3 × 3

NOISE MODEL	MARGINAL MEDIAN	R-ORDERING	C-ORDERING	VMF	RVMF
1	14.93	15.93	13.58	14.38	14.42
2	14.78	14.68	12.54	13.62	13.90
3	13.41	13.77	12.10	13.20	13.32
4	10.82	11.17	9.96	10.61	10.77
5	9.62	10.10	8.71	9.57	9.64
6	8.65	9.24	8.05	8.68	8.72

Algorithm for calculation of the arc length of the 3D filling curve

begin
compute $A = max(x_{1k}, x_{2k}, x_{3k})$
if *(A=0)* ***then*** *tkap = 0* ***endif***
if *(A=1)* ***then*** $tkap = max\_1(x_{1k}, x_{2k}, x_{3k})$
else
if *(Amod3 = 2)* ***then*** $p\_1 = mod\_2(x_{1k}, x_{2k}, x_{3k})$ ***endif***
if *(Amod3 = 0)* ***then*** $p\_1 = mod\_0(x_{1k}, x_{2k}, x_{3k})$ ***endif***
if *(Amod3 = 1)* ***then*** $p\_1 = mod\_1(x_{1k}, x_{2k}, x_{3k})$ ***endif***
$tkap = p\_1 + A^3$
endif
return tkap

Fig. 11. C-like algorithm for the computation of γ^{-1} in 3-D case.

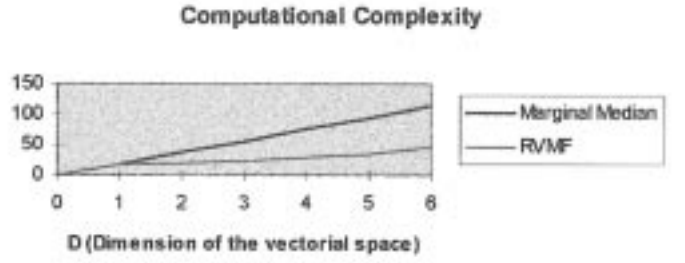


Fig. 12. Computational complexity of RVMF and marginal median.

that for a high value of D marginal median could become better.

Finally, qualitative results concerning the application of RVMF, marginal median, and r-ordering MF to three sample noisy images are presented; Lena in Fig. 13 has been corrupted by 4% impulsive noise, the squirrel image of Fig. 14 by 6% impulsive noise, and the house of Fig. 15 by 8% impulsive noise. From the visual point of view, r-ordering gives the best performances, but one can also see that performances achieved by RVMF are always comparable and in some cases (e.g., in Fig. 14) better than ones obtained by marginal median filter.

VI. CONCLUSION

In this paper, an image filtering method, called the reduced vector median filter (RVMF), has been proposed. The filter implements a new approach for extending median filters to vectorial data.

The ranking operation for multivariate data is performed by a step in which the filter operates a vectorial scalar transformation followed by scalar data ordering instead of directly realizing a median on the vectorial data. The transformation is based on the concept of a space filling curve.

The general characteristics of space filling curves have been examined and particular curves chosen for the transformation are presented for the 2-D and 3-D curves. Peculiarities of this curve have been presented and an application of the method to color image filtering has been discussed.

Results have shown the good computational efficiency of the proposed method.

Since the presented method is based on absolute sorting of vectorial data, it seems interesting to investigate possible extensions of the proposed approach to new rank-order filters and morphological operators for vectorial data processing.

APPENDIX A

In order to make easier the readability of the paper, the mathematical expressions of the terms *even* and *odd* in (28) are reported in this Appendix. At this end, let us examine the superposition of the graphical expressions of $x_{1k}(t)$ and $x_{2k}(t)$; there are four different possibilities to be considered:

Fig. 13. (a) Noisy (impulsive noise 4%) Lena image. (b) Lena image filtered by RVMF. (c) Lena image filtered by marginal median. (d) Lena image filtered by r-ordering median.

Fig. 14. (a) Noisy (impulsive noise 6%) squirrel image. (b) Squirrel image filtered by RVMF. (c) Squirrel image filtered by marginal median. (d) Squirrel image filtered by r-ordering median.

1) $\|\cdot\|_{\max} = A = x_{1k}$ and A even;
2) $\|\cdot\|_{\max} = A = x_{2k}$ and A even;
3) $\|\cdot\|_{\max} = A = x_{1k}$ and A odd;
4) $\|\cdot\|_{\max} = A = x_{2k}$ and A odd.

The *even* function is composed by contributions coming from events 1) and 2), while *odd* function is composed by contributions coming from events 3) and 4). *Even* is nonzero only if A is even, and so on for *odd*.

Fig. 15. (a) Noisy (impulsive noise 8%) house image. (b) House image filtered by RVMF. (c) House image filtered by marginal median. (d) House image filtered by r-ordering median.

Let us find how it is possible to calculate the mathematical expressions of the terms *even* is composed by. If we are in situation 1), then (22) and (24) hold. Since within the $\bar{I}$ interval only one value of t_k has to be found, the x_{2k} *profile has to be considered and, in particular, the basis function which has to be examined for* $x_{2k}(t)$ *if* $t \in \bar{I}$ is, as it is also shown by Fig. A1, $h_{l+1}(t)$, which, if $t \in \bar{I}$, holds

$$(l+1)h_1(t/(l+1)) = t \tag{A.1.1}$$

and the contribution to the *even* function given by the event 1) is then equal to x_{2k}.

By following the same reasoning for the events 2), 3), and 4), the following expressions of *even* and *odd* are obtained:

$$\text{even}(x_{1k}, x_{2k}) = \Bigg[(x_{1k} \,\mathbf{div}\, A) + (x_{2k} \,\mathbf{div}\, A) * \left(1 - \frac{(x_{1k} \,\mathbf{div}\, A)}{\frac{x_{1k}}{A} - x_{1k} \,\mathbf{mod}\, A}\right) * (2 * A - x_{1k})\Bigg]$$

$$\text{odd}(x_{1k}, x_{1k}) = \Bigg[(x_{2k} \,\mathbf{div}\, A) + (x_{1k} \,\mathbf{div}\, A) * \left(1 - \frac{(x_{2k} \,\mathbf{div}\, A)}{\frac{x_{2k}}{A} - x_{2k} \,\mathbf{mod}\, A}\right) * (2 * A - x_{2k})\Bigg].$$

The global expression that allows to compute the arc length, which is composed by a "history" term and by an "updating" term, will be then finally given by (28). Let us now demonstrate the validity of (16) for a choice of filling curve whose the arc length is expressed by (28) itself.

Let $\underline{x}_k$ and $\underline{x}_l$ be two vectors and let us impose, without a loss of generality, $||\underline{x}_k||_{\max} = A_k$ and $||\underline{x}_l||_{\max} = A_l$ with A_k, $A_l > 1$, $A_k > A_l$, $A_k = A_l + 1$. It is now possible to demonstrate that, if $t_k = \gamma^{-1}(\underline{x}_k)$ and $t_l = \gamma^{-1}(\underline{x}_l)$, $t_k > t_l$. In fact, as we have shown in this Appendix, we can write

$$\begin{aligned} t_k &= A_k^2 + \delta_k \\ t_l &= A_l^2 + \delta_l \end{aligned} \tag{A.1.2}$$

and we have $0 < \delta_k < 2 * A_k$ and $0 < \delta_l < 2 * A_l$. In the worst case, it will be, for (A.1.2), $\delta_k = 0$ and $\delta_l = 2 * A_l$. Then, we will have

$$t_k = A_k^2 = (A_l + 1)^2 \quad t_l = A_l^2 + 2 * A_l \tag{A.1.3}$$

and finally $t_k = t_l + 1 > t_l$.

This is equal to say that (16) holds for the chosen 2-D filling curve.

ACKNOWLEDGMENT

The authors wish to thank Prof. A. N. Venetsanopoulos and Dr. K. Plataniotis for useful discussions and for providing software tools for multidimensional noise generation and performance evaluation.

REFERENCES

[1] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," in *Proc. IEEE*, vol. 78, pp. 678–689, Apr. 1990.
[2] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Norwell, MA: Kluwer, 1990.
[3] T. Bially, "Space filling curves: Their generation and their application to bandwidth reduction," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 658–664, 1969.
[4] A. R. Butz, "Alternative algorithm for Hilbert's space filling curve," *IEEE Trans. Comput.*, vol. 20, pp. 424–426, Apr. 1971.
[5] J. M. Tukey, "Nonlinear (nonsuperposable) methods for smoothing data," in *Proc. Congr. EASCON*, 1974, p. 673.

[6] K. Tang, J. Astola, and Y. Neuvo, "Nonlinear multivariate image filtering techniques," *IEEE Trans. Image Processing*, vol. 4, pp. 788–798, June 1995.
[7] I. Pitas and C. Kotropoulos, "Multichannel L filters based on marginal data ordering," *IEEE Trans. Signal Processing*, vol. 42, Oct. 1994.
[8] J. Scharcanski and A. N. Venetsanopoulos, "Color edge detection using directional operators," in *Proc. 1995 IEEE Workshop Nonlinear Signal and Image Processing*, Neos Marmaras, Greece, June 1995.
[9] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Color image processing using fuzzy vector directional filters," in *Proc. 1995 IEEE Workshop on Nonlinear Signal and Image Processing*, Neos Marmaras, Greece, June 1995.
[10] C. Alexopoulos, N. Bourbakis, N. Ioannou, "Image encryption method using a class of fractals," *J. Electron. Imaging*, pp. 251–259, July 1995.
[11] M. K. Quweider and E. Salari, "Peano scanning partial distance search for vector quantization," *IEEE Signal Processing Lett.*, vol. 2, pp. 169–171, Sept. 1995.
[12] S. Kamata, A. Perez, and E. Kawaguchi, "N-dimensional Hilbert scanning and its application to data compression," in *Proc. SPIE: Vol. 1452, Image Processing Algorithms and Technology*, 1991, pp. 430–441.
[13] K. N. Plataniotis, C. S. Regazzoni, A. Teschioni, and A. N. Venetsanopoulos, "A new distance measure for vectorial rank order filters based on space filling curves," in *Proc. IEEE Conf. Image Processing*, ICIP '96, Lausanne, Switzerland, vol. 1, pp. 411–414.
[14] M. Barni, V. Cappellini, and A. Mecocci, "The use of different metrics in vector median filtering: Application to fine arts and paintings," in *Proc. EUSIPCO-92, 6th Europ. Signal Processing Conf.*, Brussels, Belgium, Aug. 25–28, 1992, pp. 1485–1488.
[15] P. Baglietto, M. Maresca, M. Migliardi, and N. Zingirian, "Image processing on high-performance RISC systems," in *Proc. IEEE*, Aug. 1996, pp. 917–930.
[16] B. Codenotti and M. Leoncini, *Fondamenti di Calcolo Parallelo*. Reading, MA: Addison-Wesley, 1990.

Carlo S. Regazzoni (S'90–M'92) was born in Savona, Italy, in 1963. He received the Laurea degree in electronic engineering and the Ph.D. degree in telecommunications and signal processing from the University of Genoa, Italy, in 1987 and 1992, respectively.

He is an Assistant Professor in Telecommunications, Department of Biophysical and Electronic Engineering (DIBE), University of Genoa, since 1995. In 1987, he joined the Signal Processing and Understanding Group (SPUG) of DIBE, where he is responsible of the Industrial Signal and Image Processing (ISIP) area since 1990. The ISIP area participated in several EU research and development projects (ESPRIT (P7809 DIMUS, P8433 PASSWORDS, P6068 ATHENA)). From 1993 to 1995, he was a visiting scientist at the University of Toronto, Toronto, Ont., Canada. His main research interests are probabilistic nonlinear techniques for signal and image processing, nonconventional signal detection and estimation techniques, and distributed data fusion in multisensor systems.

Dr. Regazzoni is a referee of several international journals and is a reviewer for EU projects (ESPRIT *BRA, 1993, LTR 1994–1997, MAST 1996*) and Italian Higher Education evaluation projects (CAMPUS 1997). He is a member of IAPR and AIIA.

Andrea Teschioni was born in Genoa, Italy, in 1969. He obtained the Laurea degree in electronic engineering from the University of Genoa in 1994, with a thesis on nonlinear and adaptive image processing for multilevel segmentation. Since 1996, he has been a Ph.D. student in electronic engineering at the University.

In 1995, he joined the research team on Signal Processing and Understanding, Department of Biophysical and Electronic Engineering, University of Genoa, where he is working in the industrial signal and image processing area. He is currently involved in the research fields of color image processing, image sequence analysis, and signal processing for mobile communications.