A NEW FAMILY OF ORDER-STATISTICS BASED SWITCHING VECTOR FILTERS

M. Emre Celebi*  
Dept. of Computer Science  
Louisiana State University  
Shreveport, LA, USA  
ecelebi@lsus.edu

Gerald Schaefer  
Dept. of Computer Science  
Loughborough University  
Loughborough, UK  
gerald.schaefer@ieee.org

Huiyu Zhou  
Queens University Belfast  
Belfast, UK  
H.Zhou@ecit.qub.ac.uk

ABSTRACT
In this paper, we present a family of order-statistics based vector filters for the removal of impulsive noise from color images. These filters preserve the edges and fine image details by switching between the identity (no filtering) operation and a robust order-statistics based filter operation based on the univariate median operator. Experiments on a diverse set of images and comparisons with state-of-the-art filters show that the proposed filters combine simplicity, flexibility, good filtering quality, and low computational requirements.

Index Terms— Noise removal, impulsive noise, nonlinear vector filter, order-statistics, switching filter

1. INTRODUCTION
Color images are often contaminated with noise, which is introduced during acquisition or transmission. In particular, the introduction of impulsive noise into an image not only lowers its perceptual quality, but also makes subsequent tasks such as edge detection and segmentation more difficult. Therefore, the removal of such noise is often an essential preprocessing step in many color image processing applications.

Numerous filters have been proposed for the removal of impulsive noise from color images [1, 2, 3, 4]. Among these, nonlinear vector filters have proved successful in the preservation of edges and fine details while removing the noise. Early approaches to nonlinear filtering of color images often involved the application of a scalar filter to each color channel independently. However, since separate processing ignores the inherent correlation between the color channels, these methods often introduce color artifacts to which the human visual system is very sensitive. Therefore, vector filtering techniques that treat the color image as a vector field and process color pixels as vectors are more appropriate. An important class of nonlinear vector filters is the one based on robust order-statistics with the vector median filter (VMF) [5], basic vector directional filter (BVDF) [6], and directional distance filter (DDF) [7] being the most widely known examples.

These filters involve reduced ordering [8] of a set of input vectors within a window to determine the output vector.

2. PROPOSED FILTER FAMILY
Consider an \( M \times N \) RGB input image \( X \) that represents a 2D array of vectors \( x(r,c) = [x_1(r,c), x_2(r,c), x_3(r,c)] \) occupying the spatial location \((r,c)\), with the row and column indices \( r = \{1,\ldots,M\} \) and \( c = \{1,\ldots,N\} \), respectively. In the pixel \( x(r,c) \), the \( x_k(r,c) \) values signify the red \((k=1)\), green \((k=2)\), and blue \((k=3)\) components. In order to isolate small image regions, each of which can be treated as stationary, an \( \sqrt{m} \times \sqrt{n} \) supporting window \( W(r,c) \) centered on pixel \( x(r,c) \) is used. The window slides over the entire image \( X \) and the procedure replaces the input vector \( x(r,c) \) with the output vector \( y(r,c) = F(W(r,c)) \) of a filter function \( F(\cdot) \) that operates over the samples inside \( W(r,c) \). Repeating the procedure for each pair \((r,c)\), with \( r = \{1,\ldots,M\} \) and \( c = \{1,\ldots,N\} \), produces output vectors \( y(r,c) \) of the \( M \times N \) filtered image \( Y \). For notational simplicity, the vectors inside \( W(r,c) \) are re-indexed as \( W(r,c) = \{x_i : i = 1,\ldots,n\} \), as commonly seen in the related literature [1, 2, 3, 4] (see Fig. 1). In this notation, the center pixel in \( W \) is given by \( x_{(n+1)/2} \) and in the vector \( x_i = [x_{i1}, x_{i2}, x_{i3}] \) with components \( x_{ik} \), the \( i \) and \( k \) indices denote the block location and color channel, respectively.

Order-statistics based vector filters operate by ranking the vectors inside the filter window using various criteria. For example, VMF uses the Minkowski distance and determines...
Fig. 1. Indexing convention inside a $3 \times 3$ window

\[
\begin{bmatrix}
    x_1 & x_2 & x_3 \\
    x_4 & x_5 & x_6 \\
    x_7 & x_8 & x_9 \\
\end{bmatrix}
\]

the lowest-ranked input vector as the output vector:

\[
y(r, c) = x_{VMF} = \arg \min_{x_i \in W(r, c)} \left( \sum_{j=1}^{n} L_p(x_i, x_j) \right)
\]

\[
L_p(x_i, x_j) = \left( \sum_{k=1}^{n} |x_{ik} - x_{jk}|^p \right)^{1/p}
\]

where $L_p$ denotes the Minkowski distance.

BVDF uses the Cosine distance to determine the output vector:

\[
y(r, c) = x_{BVDF} = \arg \min_{x_i \in W(r, c)} \left( \sum_{j=1}^{n} A(x_i, x_j) \right)
\]

\[
A(x_i, x_j) = \arccos \left( \frac{x_{i1}x_{j1} + x_{i2}x_{j2} + x_{i3}x_{j3}}{\|x_{i1}\|_2\|x_{j1}\|_2} \right)
\]

where $A(x_i, x_j)$ denotes the angle between the two vectors $x_i$ and $x_j$ and $\| \cdot \|_2$ is the $L_2$ (Euclidean) norm. On the other hand, DDF combines the Cosine and Minkowski distances:

\[
y(r, c) = x_{DDF} = \arg \min_{x_i \in W(r, c)} \left( \sum_{j=1}^{n} A(x_i, x_j) \sum_{j=1}^{n} L_p(x_i, x_j) \right)
\]

The fundamental order statistics based filters (VMF, BVDF, DDF) as well as their fuzzy and hybrid extensions share a common deficiency in that they are implemented uniformly across the image and tend to modify pixels that are not corrupted by noise. This results in excessive smoothing and the consequent blur of edges and loss of fine image details. In order to overcome this, intelligent filters that switch between the identity operation and a robust order-statistics based filter such as the VMF have been introduced [9, 10, 11, 12]. These filters determine whether the pixel under consideration is noisy or not in the context of its neighborhood. In the former case, the pixel is replaced by the output of the noise removal filter; otherwise, it is left unchanged to preserve the desired (noise-free) signal structures.

In this paper, we propose a family of switching order-statistics based vector filters defined by:

\[
y(r, c) = \begin{cases} 
    x_{(n+1)/2} & \text{if } \ d_{(n+1)/2} \leq \alpha \cdot \text{med}(d_1, \ldots, d_n) \\
    x_F & \text{otherwise}
\end{cases}
\]

where \text{med}(\cdot) is the robust univariate median operator. The Minkowski distance based member of this family (Robust Switching Vector Median Filter - RSVMF) is defined as $d_i = \sum_{j=1}^{n} L_p(x_i, x_j)$ and $x_F = x_{VMF}$. The Cosine distance based member (Robust Switching Basic Vector Directional Filter - RSBVDF) is defined as $d_i = \sum_{j=1}^{n} A(x_i, x_j)$ and $x_F = x_{BVDF}$. Finally, the directional distance based member (Robust Switching Directional Distance Filter - RSDDF) is defined as $d_i = \left( \sum_{j=1}^{n} A(x_i, x_j) \right) \left( \sum_{j=1}^{n} L_p(x_i, x_j) \right)$ and $x_F = x_{DDF}$.

The members of the filter family defined by (4) operate as follows. First, they determine whether or not the center pixel is noisy. A noisy pixel is one whose cumulative distance $d_{(n+1)/2}$ is greater than the median cumulative distance in its neighborhood. If the center pixel is noisy, it is replaced by the output of the VMF, BVDF, or DDF. Otherwise, it remains unchanged. The switching threshold can be adjusted using the $\alpha$ parameter.

The rationale behind the choice of the median operator is its statistically robust nature. In other words, this operator is resistant to noise, which makes it a suitable threshold operator. The proposed filter family utilizes this robust operator to determine whether the cumulative distance associated with the center pixel is significantly greater than a ‘typical’ cumulative distance in the neighborhood. If this is the case, the center pixel is considered to be noisy and is replaced by the VMF, BVDF, or DDF output. Otherwise, it is left unchanged to preserve the image details.

3. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we evaluate the performance of the proposed filter family on a set of test images commonly used in the color image filtering literature. In the experiments, the filtering window was set to $3 \times 3$ and the $L_2$-norm was used whenever the Minkowski distance is involved. The corruption in the images was simulated by the widely used correlated impulse noise model [13]:

\[
x = \begin{cases} 
    o & \text{with prob. } 1 - \varphi, \\
    \{r_1, o_2, o_3\} & \text{with prob. } \varphi_1 \cdot \varphi, \\
    \{o_1, r_2, o_3\} & \text{with prob. } \varphi_2 \cdot \varphi, \\
    \{o_1, o_2, r_3\} & \text{with prob. } \varphi_3 \cdot \varphi, \\
    \{r_1, r_2, r_3\} & \text{with prob. } (1 - (\varphi_1 + \varphi_2 + \varphi_3)) \cdot \varphi
\end{cases}
\]

where $o = \{o_1, o_2, o_3\}$ and $x = \{x_1, x_2, x_3\}$ represent the original and noisy color vectors, respectively, $r = \{r_1, r_2, r_3\}$ is a random vector that represents the impulse noise, $\varphi$ is the sample corruption probability, and $\varphi_1$, $\varphi_2$, and $\varphi_3$ are the corruption probabilities for the red, green, and blue channels, respectively. In the experiments, the channel corruption probabilities were set to 0.25.

Filtering performance was evaluated by the Mean Absolute Error (MAE) and Peak Signal-to-Noise Ratio (PSNR) measures [4]. The former measures the detail preservation...
Table 1. MAE comparison of the filters

<table>
<thead>
<tr>
<th>Filter</th>
<th>Airplane</th>
<th>Goldhill</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5% 10% 15%</td>
<td>5% 10% 15%</td>
</tr>
<tr>
<td>NONE</td>
<td>3.17 6.34 9.46</td>
<td>3.17 6.35 9.53</td>
</tr>
<tr>
<td>ASVMF</td>
<td>0.74 0.92 1.23</td>
<td>1.36 1.37 1.58</td>
</tr>
<tr>
<td>AVMF</td>
<td>0.49 0.99 1.51</td>
<td><strong>0.50 1.00 1.50</strong></td>
</tr>
<tr>
<td>BVDF</td>
<td>3.12 3.31 3.45</td>
<td>5.78 5.89 6.01</td>
</tr>
<tr>
<td>DDF</td>
<td>2.51 2.63 2.75</td>
<td>4.25 4.36 4.48</td>
</tr>
<tr>
<td>EVMF</td>
<td>0.72 0.88 1.18</td>
<td>1.41 1.39 1.59</td>
</tr>
<tr>
<td>RSBVDF</td>
<td>0.55 0.78 1.11</td>
<td>1.02 1.33 1.81</td>
</tr>
<tr>
<td>RSDDF</td>
<td>0.58 0.71 0.88</td>
<td>1.07 1.18 1.38</td>
</tr>
<tr>
<td>RSVMF</td>
<td><strong>0.46 0.59 0.82</strong></td>
<td>0.93 1.04 <strong>1.28</strong></td>
</tr>
<tr>
<td>VMF</td>
<td>2.45 2.58 2.70</td>
<td>4.08 4.20 4.33</td>
</tr>
</tbody>
</table>

Table 2. PSNR comparison of the filters

<table>
<thead>
<tr>
<th>Filter</th>
<th>Airplane</th>
<th>Goldhill</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5% 10% 15%</td>
<td>5% 10% 15%</td>
</tr>
<tr>
<td>NONE</td>
<td>20.91 17.92 16.18</td>
<td>21.07 18.04 16.27</td>
</tr>
<tr>
<td>ASVMF</td>
<td>36.75 34.09 31.74</td>
<td>34.41 33.37 31.57</td>
</tr>
<tr>
<td>AVMF</td>
<td>34.38 31.35 29.53</td>
<td>34.55 31.60 29.87</td>
</tr>
<tr>
<td>BVDF</td>
<td>32.17 31.37 30.79</td>
<td>28.32 28.01 27.63</td>
</tr>
<tr>
<td>DDF</td>
<td>33.81 33.34 32.84</td>
<td>31.04 30.85 30.63</td>
</tr>
<tr>
<td>EVMF</td>
<td>37.33 34.52 31.87</td>
<td>34.50 33.54 31.57</td>
</tr>
<tr>
<td>RSBVDF</td>
<td>37.00 34.15 31.51</td>
<td>33.23 30.43 27.75</td>
</tr>
<tr>
<td>RSDDF</td>
<td>37.73 35.89 33.83</td>
<td>35.65 34.60 32.95</td>
</tr>
<tr>
<td>RSVMF</td>
<td><strong>39.34 37.00 33.94</strong></td>
<td>36.21 <strong>35.35 33.28</strong></td>
</tr>
<tr>
<td>VMF</td>
<td>34.01 33.54 33.05</td>
<td>31.39 31.17 30.89</td>
</tr>
</tbody>
</table>

There is only a single parameter involved in the presented filter family, \( \alpha \). Higher values of \( \alpha \) preserve the image details better, whereas lower values remove the noise better, i.e., produce smoother results. As \( \alpha \) approaches 0, RSVMF, RSBVDF, RSDDF turn into VMF, BVDF, and DDF, respectively. In other words, maximum filtering is performed. In contrast, as \( \alpha \) approaches \( \infty \), the members of the presented filter family turn into identity filters, i.e., no filtering is performed. We have observed that \( \alpha = 1.25 \) achieves a good balance between noise removal and detail preservation in the abovementioned test image set. On highly noisy or textured images, higher \( \alpha \) values might be preferable.

We compared the proposed filter family with the following filters: VMF, BVDF, DDF, Adaptive Vector Median Filter (AVMF) [9], Entropy Vector Median Filter (EVMF) [10], and Adaptive Sigma Vector Median Filter (ASVMF) [11]. Furthermore, the identity filter (NONE) which performs no filtering was included as a baseline. Tables 1 and 2 show respectively the MAE and PSNR values for each filter on the commonly used Airplane, Goldhill, Lenna, and Peppers images. The best values are shown in bold. It can be seen that the members of the proposed filter family, particularly RSVMF, compare favorably with state-of-the-art filters.

Fig. 2 shows sample filtering results for a close-up part of the Goldhill image. It can be seen that even though the non-switching VMF suppresses the noise very well, this comes at the expense of the blurring of image details. On the other hand, the switching filters, i.e., AVMF, ASVMF, RSVMF, and RSDDF preserve the details significantly better. Among these, AVMF achieves the best detail preservation performance. However, this filter fails to remove the noise satisfactorily which is also evident by its relatively small PSNR value (see Table 2). RSVMF and RSDDF perform the best, with the former achieving a better balance between noise removal and detail preservation.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a family of switching filters for the removal of impulsive noise from color images. The proposed filters utilize the univariate median operator to switch between the identity operation and a robust order-statistics filter operation. Experiments on a diverse set of images and comparisons with state-of-the-art filters showed that the proposed filters combine simplicity, flexibility, good filtering quality, and low computational requirements.

A common problem with the current switching vector filters is that they often perform excessive smoothing in highly textured areas. Future work will be directed towards the design of adaptive switching criteria that can distinguish between a textured and a noisy neighborhood.
5. REFERENCES


Fig. 2. Filtering results for the Goldhill image