MARKOVIAN CLUSTERING FOR THE NON-LOCAL MEANS IMAGE DENOISING

Rachid Hedjam Reza Farrahi Moghaddam Mohamed Cheriet

Synchromedia Laboratory for Multimedia Communication in Telepresence
École de Technologie Supérieure, Montréal (QC), H3C 1K3 Canada
{rachid.hedjam, reza.farrahi-moghaddam, mohamed.cheriet}@ens.etsmtl.ca

ABSTRACT
The non-local means filter is one of powerful denoising methods which allows participation of far, but proper pixels in the denoising process. Although the weights of non-similar pixels are very small, high number of these pixels results in introduction of blur. In this work, we introduce an automatic and robust method to select the best candidate pixels based on their similarity to the target pixel. This method is based on graphs partitioning and uses Markovian clustering on the pixel adjacency graph (PAG). In this way, a set of relevant pixels is obtained that is used in weighted averaging for denoising each pixel. To evaluate the method, denoising of the natural images is conducted, and the results are compared to the standard NLM filter and the SVD-based method. The results are promising.

Key words: Non-Local means, Image denoising, Markov Clustering.

1. INTRODUCTION

Image denoising is a classical problem in image processing [1]. The goal is to estimate the original image \( f \) of a given observed image \( I \) using some regularization. In [2], the non-local means (NLM) image denoising algorithm is introduced based on weighted averaging and similarity of patches. The computing of similarity between patches (which are usually large, for example, \( 7 \times 7 \) pixels) is costly. In addition to standard, direct comparison, reduction of computational cost using principal component analysis (PCA) has been employed with promising results [3]. By accepting the comparison of weights based on the difference of neighboring patches around pixels, one should answer the question: what pixels should participate in the denoising process? The non-local measure of the NLM method allows a large number of pixels of the search window participate in denoising. Although this means participation of many relevant pixels, at the same time a large number of irrelevant pixels will be considered in the averaging. The weights of irrelevant pixels are usually very small, but, because of their large number, they may introduce blur. Excluding irrelevant pixels is a challenging task in improving NLM filter. In [4], another method of non-local denoising of images has been introduced based on an adaptive neighborhood selection according to a balance between the accuracy of approximation and the stochastic error, at each spatial position. Entropy-based method has also been developed for denoising in an unsupervised information-theoretic, adaptive filter (UINTA) [5]. Removing irrelevant candidate pixels based on the set of low dimensional features (obtained in PCA method) has also been used successfully [6]. Finally, an alternative strategy based on the singular value decomposition (SVD) to eliminate non-similar pairs has been used [7]. In all of these approaches, the selection process is based on some \textit{a priori} information (such as noise variance). Also, in many of them, the process is very costly (for example, the PCA-based method). In this work, a method for robust selection of the relevant pixels is proposed which is completely automatic and works only with the weights of similarity. Also, a fast implementation of the method is provided which has a small computational cost with respect to the computation of the weights step. The organization of the paper is as follows. In section 2, the basic of the NLM method is presented. The concept of pixels adjacency graph (PAG) and selection of the best candidate pixels is provided in section 3. In section 4, the Markov clustering (MCL) approach [8] for clustering the PAG is discussed. The fast implementation of the modified MCL (MMCL) is presented in section 5. Then the experiment results and comparison with other methods are provided in section 6. Finally, the conclusion and future prospects are presented.

2. NON-LOCAL MEANS DENOISING

The non-local means denoising algorithm replaces the noisy gray-values \( I(i) \) of each pixel \( i \) with a weighted average of the gray-values of all pixels on the image \( I \). We denote by \( i \) the pixel to be denoised (or target pixel) and by \( j \) the candidate pixel used to denoise \( i \). The estimate value \( \hat{I}(i) \) for a pixel \( i \) is computed based on the weighted average of all the pixels \( j \) on the image:

\[
\hat{I}(i) = \sum_{j \in N_i} w_{ij} I(j)
\]
where \( N(i) \) is the search window of size \((2s + 1) \times (2s + 1)\) centered at \( i \) and the weight \( w_{ij} \) of two pixels \( i \) and \( j \) is computed depending on the similarity of their patches and is defined as
\[
    w_{ij} = \frac{1}{Z_i} \exp \left( -\frac{1}{h} G_d * |I(N^s(i)) - I(N^s(j))|^2 \right)
\]
where \( Z \) is the normalizing term, \( Z_i = \sum_j w_{ij} \), \( G_d \) is a Gaussian spatial kernel, \( * \) is the convolution operator and \( h \) acts as a filtering parameter. It controls the decay of the exponential function and therefore the decay of the weights as a function of the Euclidean distances. This parameter is typically adjusted manually in the algorithm.

There are many dissimilar pixels in the search window. If we consider them in the averaging process (1), it may introduce blur on the denoised image. These pixels are called outliers and must be excluded from the weighting (1). In [9], a thresholding method has been proposed based on the variance of the noise in the image. In this work, an automatic clustering has been devised to select the most relevant pixels to the target pixel from the set of the candidate pixels.

3. THE PROPOSED METHOD FOR SELECTION OF THE BEST CANDIDATE PIXELS

In the set of all candidate pixels \( N(j) \) used to denoise the pixel \( i \), many of pixels are too different from the target pixel \( i \). These pixels should be regarded as outliers. In terms of the NLM filter formulation, their similarity weights to the target pixel are very small, but introduce undesired error to the denoised value of pixel \( i \); it is possible that such neighborhoods become important and form a majority. Their cumulative influence is not negligible and can influence the denoising by introducing the blur. On the other hand, the relevant pixels to the target pixel, are those whose weight is relatively large. Considering only the relevant pixels, the introduced blur will be reduced. Usually, a suitable weight threshold is used to locate the relevant pixels. In this work, instead of using a threshold value, a graph partitioning, in the form of a graph adjacency, by considering the pair-wise relations (between the patch centered on pixel \( i \) to be denoised and its neighbor patches \( j \)) is used. Therefore, the selection is completely automatic and is just based on the weight values (which are computed based on the similarity of neighboring patches). Below, the detailed definition of the pixels adjacency graph (PAG) is presented.

3.1. PIXELS ADJACENCY GRAPH

For adapting the graph clustering to the NLM filter, a pixel adjacency graph (PAG) is created. The graph is represented by \( G = (V, E) \), where \( V \) is the set of graph vertices and \( E \) is their associated edges. In this graph, each pixel in the search window \( (N^s(i)) \) is associated with a unique vertex on the graph. All vertices are connected to the centered vertex, which is associated to the pixel \( i \). The weight of the connecting edges is set to the corresponding NLM weight of the pixel \( w_{ij} \). The edges between all other pixels pairs have zero weight. The graph is represented using a similarity matrix \( W \), where \( W_{ij} \) is the weight between the \( x \)th and \( y \)th vertices. In our case, the index of the vertex associated to the target pixel \( i \) is set to 0. Therefore, the first row and the first column of \( W \) consist of the weights between the target pixel and its neighbors in the search window. The similarity matrix is symmetric, and \( W_{xx} = 1 \ \forall \ x \). Figure 1 shows a typical search window \( (N^s) \) and some of connections between the target pixel (associated to the vertex 0) and its neighbors (vertices 1, 2, 3, 4, 5, 6, etc centered on the pixels \( j \) with \( j \in N \)). In this example, the patches associated with vertices 1, 5 and 6 are actually outliers patches, while the patches 2, 3 and 4 are qualified for denoising the target pixel. These relevant pixels are obtained by clustering the graph which will be discussed in the next section.

![Fig. 1. The pixel adjacency graph (PAG). The strong connections are represented by thick lines, while the thin lines represent the weak connections.](image)

4. MARKOV CLUSTERING (MCL)

The MCL [8] is a recent, fast, and efficient clustering algorithm for graphs, which is based on random walk concept. The main idea is that a random walk in a dense cluster (i.e., a strongly connected cluster) will visit many of the nodes before leaving the cluster. Usually the random walk process has been substituted by a flow on the PAG. The flow increases on strong connections and decreases for weak connections. As a result, the flow values between different clusters vanish while the flow within the clusters still exists (see Figure 2). The strengthening and weakening of the flow is done using two simple algebraic operations on the similarity matrix (i.e., the adjacency matrix associated to the PAG). The first operation is expansion which simulates \( e \) step of the random walk [8]. It is defined as the following:
\[
    W \leftarrow W^e, \quad e \in \mathbb{Z}_{>1}
\]
Expansion operation models the spreading out of the flow. The second operation is inflation, which is, mathematically speaking, a Hadamard power followed by a diagonal scaling. Inflation models the contraction of flow; it becomes thicker in
regions of higher flow current and thinner in regions of lower flow current. The inflation operation is defined as follows

\[ W_{ij} \leftarrow W_{ij}^r, \quad \forall i, j \quad r \in \mathbb{R}_{>1.0} \]  \hspace{1cm} (4)

**Fig. 2. The MCL partitioning**

The process converges toward a partition of the graph, with a set of high-flow regions (the clusters) separated by boundaries with no flow. The value of the inflation parameter \( r \) controls cluster granularity and thus influences the number of clusters.

Starting from \( G = (V, E) \), the MCL algorithm is applied to \( W \) (see algorithm 1). The matrix \( W \) depends on the number of pixels in the \( \mathcal{N}_x(i) \). If we assume that the search window has \((2s + 1)^2\) pixels, so the size of \( W \) will be \((2s + 1)^2 \times (2s + 1)^2\), due to the comparison between the target pixel and all the other pixels in \( \mathcal{N}_x(i) \). This is the nature of the Markovian matrix of the MCL. The computational time of the method is proportional to the size of the search window. It is a bottleneck especially for large \( \mathcal{N}_x \) which are required in big images. In order to reduce the computational cost and based on the nature of the PAG discussed in the previous section, we propose to perform the MCL algorithm on a vectorial representation of \( W \) (see next section for details).

In this modification, only the three significant vectors, \( u \), \( w \) and \( v \) representing respectively the first row, the first column and the diagonal of the matrix \( W \) are considered. Therefore, the computational cost is reduced significantly. At the end of clustering, only the first row of the partitioning matrix is considered. The associate indices of the non-zero values represents the relevant pixels to the target pixel.

5. MODIFIED MARKOV CLUSTERING (MMCL)

Working with the similarity matrix that is actually a large sparse matrix, needs high computational time especially in the expansion phase (matrix multiplication). The cost for the whole image is high, because the partitioning is performed for each pixel on the image. As it has been discussed in section 4, only the first row, first column and the diagonal of the matrix of similarity \( W \) contain valuable information. Here, in order to reduce the complexity of the method, the similarity matrix is represented by the three vectors \( u \), \( w \) and \( v \). The formula used to simplify the MCL operations are presented in the algorithm MMCL. After each expansion and normalization operation, \( v_1 \) and \( w_1 \) are set equal to \( u_1 \). Also, after the normalization operation, we enforce \( u_1, v_1 \) and \( w_1 \) values to be 1. The MMCL reduces the computational time significantly without any loss in the partitioning results. The number of iterations to achieve such partitioning is about 5. Then a simple thresholding \( T_x \) on the vector \( u \) (obtained by MMCL) is done to obtain the indices of the proper pixels. In practice, the weights of outlier pixels converge very quickly to zero and the weights of significant pixels approach the value of 1. These significant pixels are used in (1). The \( \delta_{a,b} \) operator in the algorithm is the Kronecker delta, it is 1 if \( a = b \) and is 0 otherwise.

\[
\text{Algorithm: MMCL}
\]

\begin{algorithm}
for \( i = 1: \text{nb iter} \) do
  1- Expansion
    \[ u_j = u_1 u_j + u_j v_j - \delta_{j,1}(u_j w_j + u_j^2) + \delta_{j,1}(<u, u >) \]
    \[ v_j = v_j^2 + w_j u_j + \delta_{j,1}(v_j u_j - v_j v_1 - w_j u_j) \]
    \[ u_j = \frac{v_j}{\sum_k u_k}, u_1 = 1, v_j = 1 - u_j \]
  2- Inflation
    \[ u_j = u_j^r, v_j = v_j^r \]
  3- Normalization
    \[ u_j = \frac{u_j}{\sum_k u_k}, u_1 = 1, v_j = 1 - u_j \]
end for
\end{algorithm}

6. EXPERIMENTAL RESULTS AND DISCUSSIONS

6.1. Set Up

In all the experiments, we consider the following internal parameters in the proposed method. The search window and the patches are \( s = 7 \) and \( d = 3 \). We use a very small positive threshold \( T_x = 0.001 \). Because the Markov clustering converges very fast, a few number of iterations (5-10) is sufficient for clustering. The inflation parameter must be higher than 1. In most over our experience a value of \( r = 1.2 \) gives better results.

6.2. Experimental results

The proposed method was tested on the natural gray images. Our denoising results are compared to that of the original non-local means [2] and the SVD-based approach [7] in terms of PSNR. Figure 3 shows the PSNR versus the standard deviation values of the Gaussian noise for the proposed method and the original NLM method. High performance of denoising in the proposed method is obtained while computational cost also reduced because of its proper selection of the patches (Figure 3). The proposed method is almost 6 times faster. In Figure 4, the output of the original NLM and the proposed method for Camera-man (512 × 512) image are presented. As it can be seen from the figure, the output of the proposed method preserves the fine structures and reduce blur. This is mainly because of good selection of relevant pixels. The same behavior can be seen in Figure 5. The amount of introduced blur in the proposed method is reduced. In Figure
6, the performance of the proposed method is compared to the SVD-based method [7]. As it can be seen, the proposed method also shows superior performance against the SVD-based method.

![Image](image_url)

**Fig. 3.** Left: SNR versus the noise standard deviation, on Lena image ($256 \times 256$), for the NLM (dashed line) and the proposed method (continuous line). Right: Computational time ratio of the proposed method to the original NLM, $n$ is the size of image.

7. CONCLUSION

MCL-based method for the selection of the relevant candidate pixels in NLM denoising is presented. This method, which is based on the flow theory on graphs, solves automatically the problem of outlier pixels. The complexity of the method is reduced significantly thanks to the introduction of a modified MCL based on vectorial calculation. The clustering method is combined with standard NLM filter and is tested on different images with various values of noise variance using a constant $h$. The numerical results are promising and competitive with other methods. As a future prospect, adaptive filters will be investigated in order to achieve a better performance.

![Image](image_url)

**Fig. 4.** Comparison between original NLM and the proposed method. From left to right, Original image, noisy image ($\sigma = 20$), NLM result (PSNR 29.27), Proposed method (PSNR 29.41).

8. REFERENCES


