VIDEO DEBLURRING IN COMPLEX WAVELET DOMAIN USING LOCAL LAPLACE PRIOR FOR ENHANCEMENT AND ANISOTROPIC SPATIALLY ADAPTIVE DENOISING FOR PSF DETECTION

Hossein Rabbani

Department of Biomedical Engineering, Isfahan University of Medical Sciences, Isfahan, IRAN

ABSTRACT

This paper presents a new algorithm for video deblurring using frames before and after each scene as a multiframe observation from that scene. For this reason we develop the recently proposed algorithms that try to benefit from advantages of advanced denoising methods. At first the data is transformed to discrete complex wavelet transform (DCWT) and an initial estimate of clean data and point spread function (PSF) is obtained based on minimization of the energy criterion in gradient projection algorithm. In the next stage we improve the estimated clean data using a denoising method employing local Laplace prior and the estimated PSF is enhanced using an anisotropic spatially adaptive denoising procedure based on the local polynomial approximation (LPA) of blur operator and the intersection of confidence intervals (ICI) used for selection of window sizes of LPA. The mentioned procedure is repeated (in gradient projection algorithm) to obtain the appropriate estimations of PSF and clean data. Applying this technique for deblurring of video sequences produces better results in comparison with other methods.

Index Terms— deblurring, complex wavelets, denoising, blind deconvolution, video processing

1. INTRODUCTION

Usually in many imaging applications such as medical imaging, astronomical imaging, radiometry and remote sensing, images are contaminated by noise and blurred due to atmospheric turbulence, relative motion between the camera and the captured object, out of focus, etc. A popular and proper model for this acquisition systems is a linear filter model with additive noise as follows:

\[ y(k) = p(k) \ast x(k) + n(k) \]  

(1)

where for an image (after transforming image matrix to vector) with N pixels, \( \{y(k)\}_{k=1}^{N} \), \( \{p(k)\}_{k=1}^{N} \) and \( \{x(k)\}_{k=1}^{N} \) are respectively degraded image, PSF and clean image and \( \{n(k)\}_{k=1}^{N} \) is the noise that is usually modeled as an additive white Gaussian noise (AWGN).

When the PSF is known, the restored image can be obtained by \( \hat{x}^{est}(f) = \hat{y}(f) / \hat{p}(f) \) where \( \hat{\cdot} \) refers to the Fourier transformation [1] (this simple frequency domain quotient deconvolution has very poor performance in the presence of noise). To dominate ill-condition positions (e.g., when \( \hat{p}(f) \) tends to zero), the regularized inverse operator \( \hat{x}^{est}(f) = \hat{y}(f)(\hat{p}(f))^{-1} / (\hat{p}(f))^{2} + \epsilon^{2} \) is used where \( \epsilon \) is the regularization parameter [1]. However in many practical situations, the PSF is also unknown and must be estimated from the observation before (final) image restoration. In this case, we are talking about blind deconvolution or blind image restoration and till now there has been a lots of techniques in this area (e.g., see [2] for a basic survey). In many applications (e.g., confocal microscopy), we have multiple observations from an specific scene that each of them has different noise and PSF function as follows [3, 4]:

\[ y_{i}(k) = p_{i}(k) \ast x(k) + n_{i}(k), i = 1, ..., M \]  

(2)

where M is the number of observations (channels) and it has been shown that the minimum value of M to get a reliable response is 3 [2]. Also, some conditions such as co-primeness requirements (about unsimilarity between observations) are required to obtain unique response in the noise-free cases and stable estimate in the noisy cases [2, 3, 4, 5].

A group of blind deconvolution methods are based on minimum least squared (MLS) algorithm [2, 3]. For example in [2] in addition to conventional quadratic criterion, another term indicating cross-correlation between channels is also used that results in a better preservation of edges. In [6], this method is developed for misaligned images. In the recent years, there has been a main growth on image denoising algorithms. In this context, some authors have tried to benefit from the advantages of recently published denoising methods. A group of these methods after blur compensation employs a denoising method for additive color noise removal (e.g., see [7] and references there in). However in this class of techniques the blur function must be known (in the whole time) or it must be estimable truly before image restoration. Another class of these methods that are based on iterative methods, employ denoising methods in each iteration to improve the estimated PSF and clean image [5] (see Figure 1). In this paper MLS algorithm as described in [5] is used for each subband of discrete complex wavelet transform (DCWT) domain [8] to obtain an initial estimate of PSF and clean image. The main
reason of applying this algorithm in DCWT domain refers to the next step of this iterative blind deconvolution algorithm, i.e., noise reduction, which it has been shown wavelet-based denoising methods are among the best [9, 10, 11]. In fact, as we discuss in Section 2, after estimation of PSF and clean image using MLS method in DCWT domain, we improve the estimated PSF using LPA-ICI denoising method [1, 5, 12]. This method is a spatially adaptive method that uses anisotropic local window for each pixel. The estimation of clean image is also improved based on modeling of noise-free wavelet coefficients using a local Laplace prior in a Bayesian framework [11]. The enhanced PSF and estimated image are substituted again in the MLS algorithm and this procedure in the DCWT domain is iterated till convergence of parameters. Section 3 is dedicated to simulation results for video deblurring based on using frames before and after each scene as multiframe observation from an image and applying the proposed method in Section 3 for blind deconvolution of each frame. Finally, we conclude this paper in Section 4 and suggest future researches in this area.

2. PROPOSED METHOD

In this section we describe the video deblurring algorithm used for blind deconvolution of each frame based on employing frames before and after each scene as multiframe observation from an image. For this reason the MLS algorithm that produces the initial estimates of PSF and clean image is described in subsection 2.1. In subsection 2.2 we review the LPA-ICI denoising method [12] that is used for improving the estimated PSF in MLS algorithm. In subsection 2.3 we introduce the denoising method based on using maximum a posterior (MAP) estimator and local Laplace prior for enhancing the estimated image in MLS algorithm and finally we conclude our video deblurring algorithm in DCWT domain in subsection 2.4.

2.1. MLS algorithm

Usually obtaining $x(k)$ and $p_i(k)$ from the observations $y_i(k)$ results in an ill-condition problem and so conditional optimization is used to solve this problem. If $Q(x)$ and $R(p_i)$ indicate the regularizations of $x(k)$ and $p_i(k)$ respectively, the optimization problem is equal to minimization of the following criterion:

$$E(x, p_1, ..., p_M) = \sum_{i=1}^{M} \frac{\|p_i \ast x - y_i\|^2}{\sigma_i^2} + \gamma_1 Q(x) + \gamma_2 R(p_1, ..., p_M)$$

where $\|\cdot\|$ indicates L2 norm and $\sigma_i^2$ is the variance of $p_i \ast x - y_i$ that is divided to $\|p_i \ast x - y_i\|$ for normalization.

Although in the classic methods (such as proposed methods by Tichonov [13]) that quadratic regularization functions (such as $\int x^2$ or $\int \nabla x^2$) are used, we can solve (3) easily using Fourier transform, but the performance of these techniques to preserve the edges is not acceptable especially in noisy cases. Therefore, other regularization functions such as Mumford-Shah [14] and total variation (TV) [15] that uses $\int \nabla x$ were introduced. In [16], it has been shown to fulfill the co-primeness requirements $\sum_{j=1}^{M} p_i \ast y_j - p_j \ast y_i$ can be used as a regularization function where $\beta_{ij}$ is the inverse of variance of $\|p_i \ast y_j - p_j \ast y_i\|$. In this base (3) is written as bellow:

$$E(x, p_1, ..., p_M) = \sum_{i=1}^{M} \frac{1}{\sigma_i^2} \|p_i \ast x - y_i\|^2 + \gamma_1 \|x\|^2 + \gamma_2 \sum_{i=1}^{M} \|p_i\|^2 + \gamma_3 \sum_{i=1}^{M} \sum_{j=1}^{M} \beta_{ij} \|p_i \ast y_j - p_j \ast y_i\|$$

This equation in the Fourier domain is rewritten as follows:

$$E(x, p_1, ..., p_M) = \sum_{i=1}^{M} \frac{1}{\sigma_i^2} \|\hat{p}_i(f) \hat{x}(f) - \hat{y}_i(f)\|^2 + \gamma_1 \|\hat{x}(f)\|^2 + \gamma_2 \sum_{i=1}^{M} \|\hat{p}_i(f)\|^2$$

$$+ \gamma_3 \sum_{i=1}^{M} \sum_{j=1}^{M} \beta_{ij} \|\hat{p}_i(f) \hat{y}_j(f) - \hat{p}_j(f) \hat{y}_i(f)\|^2$$

Using Gradient method, the iterative equations would be:

$$\hat{x}^{(j)} = \hat{x}^{(j-1)} - a_j \frac{\partial E}{\partial \hat{x}}|_{\hat{x}=\hat{x}^{(j-1)}}$$

$$\hat{p}_i^{(j)} = \hat{p}_i^{(j-1)} - b_j \frac{\partial E}{\partial \hat{p}_i} |_{\hat{p}_i=\hat{p}_i^{(j-1)}}$$

According to the Newton method [17], we can improve the convergence rate using Hessian matrix. So, dividing $\frac{\partial^2 E}{\partial \hat{x}^2}$ and $\frac{\partial^2 E}{\partial \hat{p}_i^2}$ respectively to $a_j$ and $b_j$, and after some simplification we can write:

$$\hat{x}^{(j)} = \hat{x}^{(j-1)} + a_j \frac{\sum_{i=1}^{M} \hat{y}_i \hat{p}_i^{(j-1)} + \sum_{i=1}^{M} |p_i^{(j-1)}|^2}{\gamma_1 \sigma_i^2 + \sum_{i=1}^{M} |p_i^{(j-1)}|^2}$$

$$\hat{p}_i^{(j)} = \hat{p}_i^{(j-1)} + b_j \frac{\hat{y}_i \hat{x}^{(j-1)} + \gamma_3 \hat{y}_j \sum_{i=1, t \neq j}^{M} \beta_{it} \hat{p}_t^{(j-1)} \hat{y}_t^{2}}{\gamma_2 \sigma_i^2 + \gamma_3 \sum_{i=1, t \neq j}^{M} \beta_{it} \hat{y}_t^{2}}$$
where $\beta_{jt}^{(j-1)}$ is the inverse of variance of $\| p_{jt} * y_{jt} - p_{jt} * y_{jt} \|$ for $\hat{p}_{jt} = \hat{p}_{jt}^{(j-1)}$.

2.2. LPA-ICI denoising method

In [12], an image denoising algorithm is introduced based on using anisotropic local windows. For this reason the linear directional filters $g_{n,\theta}$ are obtained using local polynomial approximation (LPA) where $\theta$ shows the direction of filter and $h$ is the length of filter in this direction. The direction of filter $\theta$ is a member of set $\{\theta_1, \theta_2, ..., \theta_L\}$. In this paper the number of directions $L$ is set to 4 (i.e., the set is $\{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$). For the noisy observation $s(k)$, to compute $z_{h,\theta}^{st}(k) = g_{h,\theta}(k) * s(k)$, for each $\theta$ the appropriate value of $h$ is selected from the increasing set $\{h_1, h_2, ..., h_J\}$. For this reason, the nonlinear intersection of confidence intervals (ICI) rule is proposed. According to ICI rule, the estimated $h$ (shown with $h^+(k, \theta)$) is the largest $h$ from the increasing set $\{h_1 < h_2 < ..., < h_J\}$ provided that the estimated data using $h^+(k, \theta)$ doesn’t have noticeable difference with the estimated data with smaller $h$’s. In this context, the confidence intervals $C_v = [z_{v,\theta}^{st}(k) - R\sigma_{z_{v,\theta}^{st}}(k), z_{v,\theta}^{st}(k) + R\sigma_{z_{v,\theta}^{st}}(k)]$ are defined (R is the smoothing parameter that we set it to 0.9 in this paper and $\sigma_{z_{v,\theta}^{st}}$ is the variance of $z_{h,\theta}^{st}$) and $D_v$ is obtained using $D_v = \bigcap_{i=1}^{V} C_i$. The largest $v$ that leads to an unempty value is called $v^+$ and so $h^+(k, \theta)$ is calculated by $h^+(k, \theta) = h_{v^+}$.

2.3. Bayesian denoising using local Laplace prior

A main group of wavelet-based image denoising method perform in a Bayesian framework [9, 10, 11]. For example maximum a posterior (MAP) can be used as a criterion and so according to the proposed probability density function (pdf) for noise and noise-free data, the produced thresholding function is used for noise removal. In many researches in this arena (e.g., see [10, 11]), it has been shown in addition to highly-tailed property of marginal distribution of wavelet coefficients (and other sparse transforms), incorporating the intrascale dependency property of spatial adjacent in denoising procedure impressively improve the results. In this base in [11] a local Laplace pdf is used as a priori distribution and for AWGN the following local shrinkage function is obtained:

$$\hat{w}(k) = \text{sign}(s(k)) \max(s(k) - \frac{\varsigma^2 \sqrt{2}}{\sigma(k)}, 0)$$

(10)

where $\hat{w}(k)$, $s(k)$, $\varsigma$, $\sigma(k)$ are respectively estimated data, noisy data, noise variance and variance of noise-free data (for $k^{th}$ data in each subband).

2.4. Video deblurring algorithm

According to the presented topics in subsections 2.1-2.3, we conclude our video deblurring in Table I.

### Table I. Proposed Video Deblurring Algorithm in This Paper

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Transformation</td>
<td>Apply DCWT to each frame</td>
</tr>
<tr>
<td>2. For each subband of current frame, propose the corresponding subband in previous and next frames as multiframe observation from a scene</td>
<td></td>
</tr>
<tr>
<td>3. Initialization: $j = 1$</td>
<td></td>
</tr>
<tr>
<td>The Gaussian function is proposed for $p_{jt}^{(0)}(k)$ (initial estimate of blur function)</td>
<td></td>
</tr>
<tr>
<td>The clean image is estimated using $x^{(0)}(k) = \frac{1}{M} \sum_{i=1}^{M} y_i(k)$</td>
<td></td>
</tr>
<tr>
<td>4. Clean data estimation: Obtain $\hat{x}^j(k)$ using (8)</td>
<td></td>
</tr>
<tr>
<td>5. Improvement of estimated data: Improve the estimated clean-data by applying proposed denoising method in subsection 2.3 to the output of previous step</td>
<td></td>
</tr>
<tr>
<td>6. Estimation of PSF: Obtain $\hat{p}_j^j(k)$ using (9)</td>
<td></td>
</tr>
<tr>
<td>7. Improvement of estimated PSF: Improve the estimated PSF by applying proposed denoising method in subsection 2.2 to the output of previous step</td>
<td></td>
</tr>
<tr>
<td>8. Iteration: $j \leftarrow j + 1$ and go to fourth step and iterate the algorithm till convergence (or till $j = J$)</td>
<td></td>
</tr>
<tr>
<td>9. Inverse Transformation: Apply inverse DCWT to the recovered subband of each frame</td>
<td></td>
</tr>
</tbody>
</table>

3. SIMULATION RESULTS

To evaluate our video deblurring method, we apply the proposed algorithm in Table I to a real $320 \times 240 \times 15$ video data. To apply DCWT we use the proposed filters in [18] for three scales. We propose the support size 15 for estimating PSF and to improve the estimated PSF (using LPA-ICI denoising algorithm) we use $H = \{1, 1, 2, 2, 3, 3, 5, 5, 7, 7, 11, 11\}$ and zero-order LPA with uniform window for $g_{h,\theta}$ is employed (so for each data, the average of observation in the window is proposed). The maximum iteration $J$ is set to 20 and $\sigma_1, \sigma_2, \sigma_3$ are set to 0.9, 0.6, $10^{-6}$, $10^{-7}$, 1.2 respectively [5].

Figure 2 shows one frame of deblurred video with proposed method in this paper. To see the effect of iteration and number of scales, we can also see in this figure the results of using 30 iterations instead of 20 iterations and one scales instead of three scales. The elapsed time for implementation of written codes (for each frame using 3 scales and 10 iterations) in MATLAB (R2008b) employing a desktop with Intel (R) Core (TM) 2 Duo CPU 7400 @ 2.80 GHz and 4GB of RAM is about 0.45 minutes. To better evaluate the results of algorithm, we degrade $512 \times 512$ grayscale Boat image with 3 different PSFs: 1) $9 \times 9$ uniform, 2) $7 \times 7$ (45° rotated) uniform, and 3) $7 \times 1$ $7 \times 7$ with support area $7 \times 7$, and add noise with blurred signal-to-noise ratio 40dB to data. The results of restored image using these three observations and applying our deblurring method can be seen in Figure 3. To evaluate the objective quality of our algorithm, we compare our method with others using peak signal-to-noise ratio (PSNR) criterion. Table II shows a comparison between our algorithm and proposed methods in [19, 20, 21] for a sample ultrasound image.
The proposed deblurring technique in this paper minimizes an energy (data term plus regularization terms) and uses the iterative Newton algorithm to compute the image and PSF that minimize the energy. Then, it proposes to refine, at each step of the Newton algorithm, the estimates of the image and the PSF, each one of them with an appropriate denoising technique. To justify why the refinement steps are important and show the advantage of the proposed combination of algorithms compared to just minimizing the proposed energy, we can see a comparison between deblurring results with our method and the results of deblurring without refinement steps 5 and 7 explained in Table I (for 3 scales and 10 iterations) in Figure 4. Note that by increasing iterations, quality of images will be improved and according to this figure it’s clear that using refinement steps expedites this process.

4. CONCLUSION

This paper presents a new video deblurring method in DCWT domain. An initial estimate of each frame and PSF is obtained using iterative gradient algorithm. However in each step of this algorithm we try to benefit from the advantages of recently developed spatially adaptive denoising methods to enhance the estimations. We use LPA-ICI denoising method to improve the estimated PSF in each iteration and denoising method based on local Laplace prior in a Bayesian framework to enhance the estimated clean data.

Since the computational complexity is a key issue for video coding, we are trying to improve the speed of this algorithm using faster denoising methods than LPA-ICI and obtain better results by applying the proposed method in more appropriate transforms for video processing such as 3D DCWT or in a real codec such as JPEG2000.

5. REFERENCES


Table I. Comparison between our algorithm and other methods in terms of PSNR for a sample ultrasound image degraded with various blurred signal-to-noise ratio (BSNR)

<table>
<thead>
<tr>
<th>BSNR</th>
<th>Proposed Method in [19, 20, 21]</th>
<th>Non-blind</th>
<th>Our Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>21.71</td>
<td>19.45</td>
<td>23.19</td>
</tr>
<tr>
<td>20</td>
<td>21.49</td>
<td>19.85</td>
<td>25.12</td>
</tr>
<tr>
<td>30</td>
<td>21.79</td>
<td>19.94</td>
<td>29.34</td>
</tr>
<tr>
<td>40</td>
<td>21.84</td>
<td>19.97</td>
<td>31.90</td>
</tr>
</tbody>
</table>