

VIDEO DENOISING USING HIGHER ORDER OPTIMAL SPACE-TIME ADAPTATION

Hae Jong Seo and Peyman Milanfar

Electrical Engineering Department, University of California at Santa Cruz
1156 High Street, Santa Cruz, CA, 95064
{rokaf,milanfar}@soe.ucsc.edu

ABSTRACT

The optimal spatial adaptation (OSA) method [1] proposed by Boulanger and Kervrann has proven to be quite effective for spatially adaptive image denoising. This method, in addition to extending the Non-Local Means(NLM) method of [2], employs an iteratively growing window scheme, and a local estimate of the mean square error to very effectively remove noise from images. By adopting an iteratively growing *space-time* window, the method was recently extended to 3-D for video denoising in [3]. In the present paper, we demonstrate a simple, but effective improvement on the OSA method in both 2- and 3-D. We demonstrate that the OSA implicitly relies on a *locally constant* model of the underlying signal. Thereby, removing this constraint and introducing the possibility of higher order local regression models, we arrive at a relatively simple modification that results in an improvement in performance. While this improvement is observed in both 2-D and 3-D, we concentrate on demonstrating it in 3-D for the application of video denoising.

Index Terms— video denoising, regression, patch-based restoration

1. INTRODUCTION

Recently, the so-called Non-Local Means method (NLM) has been proposed by Buades *et al.* [2] where a weighed averaging scheme was used to perform image denoising by making use of the fact that in natural images many structural similarities are present in different parts of the image. Since its advent, the idea of NLM for denoising purposes has inspired a significant number of researchers to modify and improve NLM. The so-called optimal spatial adaptation (OSA) method proposed by Kervrann *et al.*[1] is one such example, which improves upon the earlier NLM by iteratively growing the local analysis window size along with a pixel-wise local stopping rule based on a local estimate of the mean square error [1]. OSA was further extended to 3-D [3] for video denoising by applying iteratively growing *space-time* window, which at the time of its publication, apparently achieved state-of-art

video denoising performance. In this paper, we demonstrate a simple, but effective improvement on the OSA method in both 2- and 3-D. We demonstrate that the OSA implicitly relies on a *locally constant* regression model of the underlying signal. Thereby, removing this constraint and introducing the possibility of higher order local regression models, we arrive at a relatively simple modification that results in significant improvement in performance. In Section 2, we review the kernel regression framework in 2-D and 3-D and show how OSA in 2-D and 3-D can be extended by kernel regression using higher order signal models. In Section 3, we examine and demonstrate its performance in video denoising. Finally, Section 4 summarizes the contributions of this paper.

2. HIGHER ORDER OPTIMAL SPACE-TIME ADAPTATION APPROACH

2.1. Kernel Regression Framework

In the kernel regression framework proposed by Takeda *et al.*[4], the data model is defined for data in N -D as

$$y_i = z(\mathbf{x}_i) + \varepsilon_i, \quad i = 1, \dots, P, \quad \mathbf{x}_i = [x_{1i}, x_{2i}, \dots, x_{Ni}]^T \quad (1)$$

where y_i is a noisy sample at \mathbf{x}_i , $z(\cdot)$ is the (hitherto unspecified) *regression function* to be estimated, ε_i is an i.i.d zero mean noise, P is the total number of samples in a neighborhood (window) of interest. In this paper, we are interested in the case of $N = 3$ for the most part, as this relates to processing of video.

While the particular form of $z(\cdot)$ is unspecified, we use a generic expansion of the function about a sampling point \mathbf{x} as a model within the window of interest. Specifically, we have the, say, M -th order Taylor series

$$\begin{aligned} z(\mathbf{x}_i) &\approx z(\mathbf{x}) + \{\nabla z(\mathbf{x})\}^T (\mathbf{x}_i - \mathbf{x}) \\ &\quad + \frac{1}{2} (\mathbf{x}_i - \mathbf{x})^T \{\mathcal{H}z(\mathbf{x})\} (\mathbf{x}_i - \mathbf{x}) + \dots \quad (2) \\ &= \beta_0 + \beta_1^T (\mathbf{x}_i - \mathbf{x}) + \beta_2^T \text{vech}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\} + \dots \quad (3) \end{aligned}$$

where ∇ and \mathcal{H} are the gradient (2×1) and Hessian (2×2) operators, respectively, and $\text{vech}(\cdot)$ is the half-vectorization operator which lexicographically orders the lower triangular portion of a symmetric matrix.

This work was supported in part by AFOSR Grants F49620-03-1-0387 and FA9550-07-1-0365.

With this generic model, we wish to estimate the unknown parameters $\{\beta_n\}_{n=0}^M$ of the model from the given data. It is worth noting that in the end, we are only really interested in the first unknown β_0 , which represents the estimate of the pixel of interest. Nevertheless, considering the higher order model is useful as it provides a mechanism for capturing more complex behavior and relationships between similar pixels both spatially and radiometrically.

In order to solve for the unknowns of interest, one may naturally pose a *weighted* least squares formulation:

$$\min_{\{\beta_n\}_{n=0}^M} \sum_{i=1}^P \left[y_i - \beta_0 - \beta_1^T (\mathbf{x}_i - \mathbf{x}) - \beta_2^T \text{vech} \left\{ (\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T \right\} - \dots \right]^2 \cdot K(\mathbf{x}_i - \mathbf{x}, y_i - y), \quad (4)$$

where the *kernel* function $K(\mathbf{x}_i - \mathbf{x}, y_i - y)$ gives higher weight to measured pixels that are in some way more similar to the center pixel in the analysis region.

A special case of the above formulation is the NLM or OSA frameworks. In particular, if we set the order of the regression (M) to zero (that is, consider a 0^{th} order model), and use the non-local exponentially weighted similarity functions for $K(\cdot, \cdot)$, as suggested in the work of Buades *et al.* [2], we arrive at the NLM formulation. To be more specific, given data points $y_i, i = 1, \dots, P$, the standard NLM procedure results from the *weighted* least squares formulation which leads to

$$\hat{\beta}_0 = \hat{z}(\mathbf{x}_i) = \frac{\sum_{i=1}^P y_i K_{NLM}(\mathbf{x}_i - \mathbf{x}, y_i - y)}{\sum_{i=1}^P K_{NLM}(\mathbf{x}_i - \mathbf{x}, y_i - y)}. \quad (5)$$

An iterative version of this, as elaborated below, with the adaptive window size, is the basis of the superior OSA formulation of Boulanger *et al.* in [1, 3]. It is important to point out that the application of the OSA algorithm in 3-D for the purpose of video denoising does not require explicit motion estimation, as also indicated in [5, 6]. The same comment applies to the extensions of OSA which we propose in this paper. Indeed, in this paper, we argue that by using a higher order model ($M > 0$), improvements to the OSA algorithm can be arrived at with relatively little additional computational cost.

2.2. Higher Order Optimal Spatial Adaptation Method

The key idea behind the (standard, 0^{th} order) OSA in both 2-D and 3-D is to iteratively grow the size of a local search window Δ_i starting with a small size at each pixel and to stop the iteration at an “optimal” window size. The dimensions of the search window in 2-D grow as $(2^\ell + 1) \times (2^\ell + 1)$ where ℓ is the number of iterations while, in the 3-D case, the spatial and temporal extent are alternately increased until a related stopping rule is satisfied at each iteration. To be more

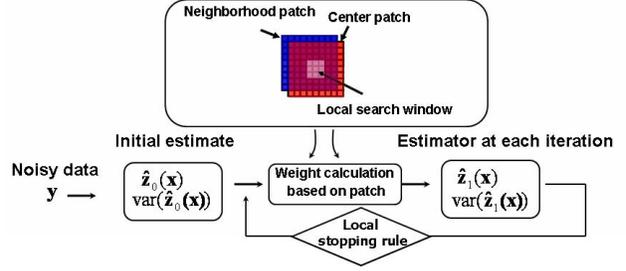


Fig. 1. Block diagram of the Optimal Spatial Adaptation

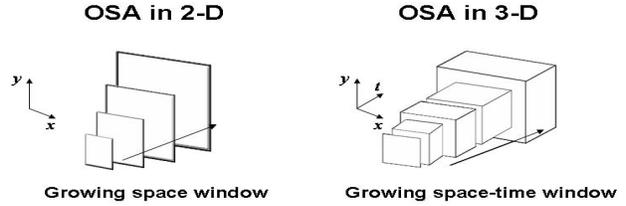


Fig. 2. Iteratively growing local search window in 2D and 3D

specific, suppose that $\hat{z}^{(0)}(\mathbf{x}_i)$ and $\hat{v}_i^{(0)}$ are the initial estimates of the pixel value and the local noise variance at \mathbf{x}_i , which are initialized as

$$\hat{z}^{(0)}(\mathbf{x}_i) = y_i, \quad \hat{v}_i^{(0)} = \hat{\sigma}^2, \quad (6)$$

where $\hat{\sigma}$ is an estimate of the noise standard deviation. (Fig. 1 illustrates a block diagram of OSA.) In each iteration, the estimate of each pixel is updated using the noisy signal and patch-based weights calculated from the previous estimate as follows:

$$\hat{z}_0^{(\ell+1)} = \hat{z}^{(\ell+1)}(\mathbf{x}_i) = \frac{\sum_{i=1}^P y_i K_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)})}{\sum_{i=1}^P K_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)})}, \quad (7)$$

where $\hat{\mathbf{z}}_{p_i}^{(\ell)}$ and $\hat{\mathbf{z}}_c^{(\ell)}$ are column-stacked vectors that contain the estimated pixels in a neighborhood patch p_i and the center patch of interest at ℓ^{th} iteration respectively, $\mathbf{H}^{(\ell)} = h_r (\hat{\mathbf{V}}^{(\ell)})^{-\frac{1}{2}}$, and h_r is the smoothing parameter. The matrix $\hat{\mathbf{V}}^{(\ell)}$ contains the harmonic means of estimated local noise variances:

$$\hat{\mathbf{V}}^{(\ell)} = \frac{1}{2} \text{diag} \left[\frac{(\hat{v}_{p_i,1}^{(\ell)})^2 (\hat{v}_{c,1}^{(\ell)})^2}{(\hat{v}_{p_i,1}^{(\ell)})^2 + (\hat{v}_{c,1}^{(\ell)})^2}, \dots, \frac{(\hat{v}_{p_i,J}^{(\ell)})^2 (\hat{v}_{c,J}^{(\ell)})^2}{(\hat{v}_{p_i,J}^{(\ell)})^2 + (\hat{v}_{c,J}^{(\ell)})^2} \right], \quad (8)$$

where J is the total number of pixels in a patch p_i , and K is defined as the Gaussian *kernel* function:

$$K_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)}) = \exp \left\{ - \frac{(\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)})^T (\hat{\mathbf{V}}^{(\ell)})^{-1} (\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)})}{h_r^2} \right\}. \quad (9)$$

A patch size J is selected so as to be able to capture the local geometry and texture in the image or video and is generally fixed at a relatively small size (5×5 or 7×7 in 2-D or $3 \times 3 \times 3$)

while the size of a local search window Δ_i grows iteratively, described in Fig. 2, a point-wise statistically-based stopping rule is applied to determine when to stop the algorithm.

What we do next is to extend the above framework to the case where the local model has higher order ($M > 0$). As such, as we alluded to earlier, the optimization problem (4) associated with kernel (9) can be expressed in matrix form as a *weighted* least-squares optimization problem,

$$\begin{aligned}\hat{\mathbf{b}} &= \arg \min_{\mathbf{b}} \|\mathbf{y} - \mathbf{H}\mathbf{b}\|_{\mathbf{W}_d}^2 \\ &= \arg \min_{\mathbf{b}} (\mathbf{y} - \mathbf{H}\mathbf{b})^T \mathbf{W}_d (\mathbf{y} - \mathbf{H}\mathbf{b}),\end{aligned}\quad (10)$$

where

$$\mathbf{y} = [y_1, y_2, \dots, y_p]^T, \quad \mathbf{b} = [\beta_0, \beta_1^T, \dots, \beta_M^T]^T, \quad (11)$$

$$\begin{aligned}\mathbf{W}_d &= \text{diag}[K_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}^{(\ell)}(\mathbf{x}_1 - \mathbf{x}) - \hat{\mathbf{z}}^{(\ell)}(\mathbf{x})), \\ &\quad K_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}^{(\ell)}(\mathbf{x}_2 - \mathbf{x}) - \hat{\mathbf{z}}^{(\ell)}(\mathbf{x})), \dots, \\ &\quad K_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}^{(\ell)}(\mathbf{x}_P - \mathbf{x}) - \hat{\mathbf{z}}^{(\ell)}(\mathbf{x}))],\end{aligned}\quad (12)$$

and

$$\mathbf{H} = \begin{bmatrix} 1 & (\mathbf{x}_1 - \mathbf{x})^T & \text{vech}^T\{(\mathbf{x}_1 - \mathbf{x})(\mathbf{x}_1 - \mathbf{x})^T\} & \dots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (\mathbf{x}_P - \mathbf{x})^T & \text{vech}^T\{(\mathbf{x}_P - \mathbf{x})(\mathbf{x}_P - \mathbf{x})^T\} & \dots \end{bmatrix}\quad (13)$$

with ‘‘diag’’ defining a diagonal matrix. Regardless of the regression order (M), since our primary interest is to compute an estimate of the image or video (pixel values), the necessary computations are limited to the ones that estimate the parameter β_0 . Therefore, the *weighted* least-square estimation is simplified to

$$\hat{z}(\mathbf{x}_i) = \mathbf{e}_1^T (\mathbf{H}^T \mathbf{W}_d \mathbf{H})^{-1} \mathbf{H}^T \mathbf{W}_d \mathbf{y}_i, \quad (14)$$

where \mathbf{e}_1 is a column vector whose elements are all zero except for the first element equal to one. This higher order formulation of OSA relaxes the implicit piecewise-constancy assumption by the choice of $M > 0$. For the sake of completeness, the overall pseudo-code for the general OSA algorithm described above is given here.

3. EXPERIMENTS

In this section, we compare our proposed higher order OSA method with standard OSA in 3-D, for the purpose of video denoising.

We selected a set of six different image sequences. Each of the six test sets are corrupted with different noise levels. The peak signal-to-noise ratio defined as $\text{PSNR} = 10 \log_{10}(255^2/\text{MSE})$ was used to measure the quality of the denoised result $\hat{\mathbf{z}}$ versus the original video \mathbf{z} . For a fair evaluation, we set the parameters in the same way as [3]. In Tables 1 and 2, we present the PSNR(dB) comparisons of the proposed higher order OSA and the standard OSA for a few sequences in cases

Algorithm 1 OSA algorithms in 3D

Let $\{J : \text{Patch size}, \alpha : \text{Percentile}, \varrho : \text{Parameter for stopping rule}, L_\Delta : \text{Maximum Iteration}\}$ be the parameters.

Initialization : compute $\hat{\sigma}^2, \hat{z}_i^{(0)}, \hat{v}_i^{2(0)}$ for each $\mathbf{x}_i \in \mathbf{G}^3$

Repeat

for every pixel $\mathbf{x}_i \in \mathbf{G}^3$ **do**

→ compute

$$\mathbf{W}_i^{(\ell)} = \frac{\mathbf{K}_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)})}{\sum_{i=1}^P \mathbf{K}_{\mathbf{H}^{(\ell)}}(\hat{\mathbf{z}}_{p_i}^{(\ell)} - \hat{\mathbf{z}}_c^{(\ell)})}$$

→ columnstack weight matrix $\mathbf{W} = \mathbf{W}_i^{(\ell)}(:)$

→ $\mathbf{W}_d = \text{diag}(\mathbf{W})$

$$\mathbf{H}_1 = [\mathbf{1}^{(\ell)} \quad \mathbf{x}_1^{(\ell)} \quad \mathbf{x}_2^{(\ell)} \quad \mathbf{x}_3^{(\ell)}]$$

$$\mathbf{H}_2 = [\mathbf{1}^{(\ell)} \quad \mathbf{x}_1^{(\ell)} \quad \mathbf{x}_2^{(\ell)} \quad \mathbf{x}_3^{(\ell)} \quad \mathbf{x}_1^{2(\ell)} \quad \mathbf{x}_2^{2(\ell)} \quad \mathbf{x}_3^{2(\ell)}]$$

$$\mathbf{x}_1^{(\ell)} \quad \mathbf{x}_2^{(\ell)} \quad \mathbf{x}_1^{(\ell)} \quad \mathbf{x}_3^{(\ell)} \quad \mathbf{x}_2^{(\ell)} \quad \mathbf{x}_3^{(\ell)}]$$

if mode = ‘‘Zeroth Order’’ **then**

$$\mathbf{K}_{\text{equi}} = \mathbf{W}_d$$

else {mode = ‘‘First Order’’}

$$\mathbf{K}_{\text{equi}} = (\mathbf{H}_1^T \mathbf{W}_d \mathbf{H}_1)^{-1} \mathbf{H}_1^T \mathbf{W}_d$$

else {mode = ‘‘Second Order’’}

$$\mathbf{K}_{\text{equi}} = (\mathbf{H}_2^T \mathbf{W}_d \mathbf{H}_2)^{-1} \mathbf{H}_2^T \mathbf{W}_d$$

end if

$$\hat{z}_i^{(\ell)} = \mathbf{e}_1^T \mathbf{K}_{\text{equi}} \mathbf{y}_i$$

$$\hat{v}_i^{2(\ell)} = \hat{\sigma}^2 \mathbf{e}_1^T \mathbf{K}_{\text{equi}} \mathbf{K}_{\text{equi}}^T \mathbf{e}_1$$

→ test the window using

$$\hat{\Delta}(\mathbf{x}_i) = \arg \max_{\Delta_i^{(\ell)} \in L_\Delta} \{|\Delta_i^{(\ell)}| : |\hat{z}_i^{(\ell)} - \hat{z}_i^{(\ell')}| < \varrho \hat{v}_i^{(\ell')}\}$$

for all $1 \leq \ell' < \ell$.

If this rule is violated at iteration ℓ , $\hat{z}_i^{(\ell-1)}$ is accepted as the final estimate at \mathbf{x}_i .

→ increment ℓ

While $\ell \leq L_\Delta$

end for

where patch size is fixed spatially as 5×5 and 7×7 respectively. The PSNR per frame was measured and averaged over the 20 middle frames of the sequence.

Our proposed method (1st order OSA) consistently outperforms the standard OSA in these tests. The key idea behind the local stopping rule in OSA is to find the smallest estimator variance among the various iterations. Through compensation of the bias-variance tradeoff resulting from employment of higher order regression, we implicitly find the estimator which minimizes the local mean square error(MSE). Besides, considering the computational complexity of the standard OSA, the fact that 1st order OSA with a 5x5 size of patch even performs better than the standard OSA with a 7x7 size of patch for some video sequences indicates that our proposed method is more efficient than the standard OSA. However, we point out for some examples that 2nd order OSA did not improve the standard OSA since the extent of variance in-

crease was larger than the extent of bias decrease in 2^{nd} order OSA($M = 2$), which led to the increase of local MSE.

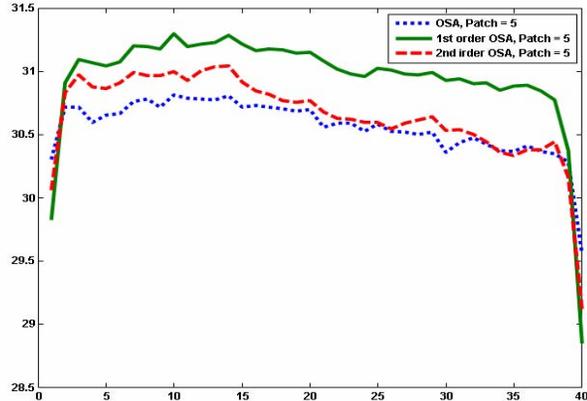


Fig. 3. Per-frame PSNR comparison of the proposed higher order OSA with standard OSA(patch=5x5) on Coastguard Sequence

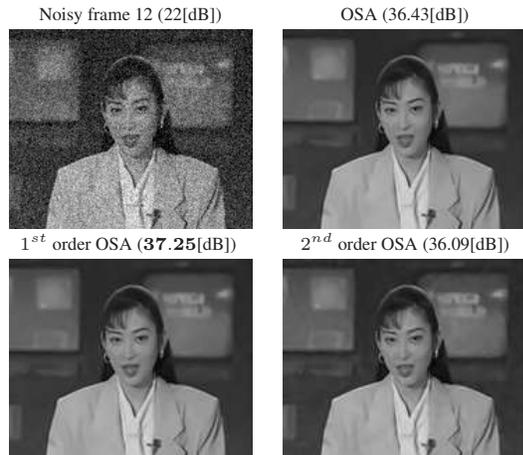


Fig. 4. Fragment of the 12th frame of Akiyo sequence denoised by the standard OSA, 1^{st} order OSA, and 2^{nd} order OSA; the noise level($\sigma = 20$). The output PSNR(for this frame only) is given in parentheses for each of the methods

4. CONCLUSION

We have presented a simple, but effective higher order extension of OSA in 2- and 3-D and verified that our proposed method results in improved performance. Our main contribution consists of achieving an improvement upon standard OSA by overcoming the implicit locally constant regression model which it employs.

5. REFERENCES

[1] J. Boulanger and C. Kervrann, "Optimal spatial adaptation for patch-based image denoising," *IEEE Transactions on Image Processing*, vol. 15, pp. 2866–2878, 2006.

| Test Sequence | Input [dB] | OSA [dB] | 1^{st} OSA [dB] | 2^{nd} OSA [dB] |
|--------------------|------------|----------|-------------------|-------------------|
| Patch = 5x5 | | | | |
| Akiyo | 22 | 36.35 | 37.15 | 36.08 |
| Garden | 28 | 29.76 | 30.37 | 29.92 |
| Miss America | 28 | 41.51 | 41.80 | 41.36 |
| Suzie | 24 | 36.50 | 36.91 | 36.51 |
| | 28 | 38.35 | 38.66 | 38.44 |
| Salesman | 24 | 33.66 | 33.89 | 33.49 |
| | 28 | 36.08 | 36.21 | 35.78 |
| Coastguard | 22 | 30.63 | 31.10 | 30.75 |

Table 1. The performance comparison between standard OSA and higher order OSAs on 6 test sequences(Average PSNR [frame: 11 ~ 30], Patch size : 5x5)

| Test Sequence | Input [dB] | OSA [dB] | 1^{st} OSA [dB] | 2^{nd} OSA [dB] |
|--------------------|------------|----------|-------------------|-------------------|
| Patch = 7x7 | | | | |
| Akiyo | 22 | 36.74 | 37.16 | 35.92 |
| Garden | 28 | 30.16 | 30.66 | 30.26 |
| Miss America | 28 | 41.75 | 41.76 | 41.28 |
| Suzie | 24 | 36.98 | 37.08 | 36.61 |
| | 28 | 38.88 | 38.92 | 38.61 |
| Salesman | 24 | 33.96 | 33.98 | 33.42 |
| | 28 | 36.34 | 36.35 | 35.61 |
| Coastguard | 22 | 31.11 | 31.44 | 30.89 |

Table 2. The performance comparison between standard OSA and higher order OSAs on 6 test sequences(Average PSNR [frame: 11 ~ 30], Patch size : 7x7)

[2] A. Buades, B. Coll, and J. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling and Simulation: A SIAM Interdisciplinary J.*, vol. 4, pp. 490–530, 2005.

[3] J. Boulanger, C. Kervrann, and P. Bouthemy, "Space-time adaptation for patch-based image sequence restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1096–1102, Jun 2005.

[4] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, pp. 349–366, Feb 2007.

[5] A. Buades, B. Coll, and J. Morel, "Denoising image sequences does not require motion estimation," *IEEE International Conference on Advanced Video and Signal Based Surveillance*, vol. 29, pp. 1096–1102, Jun 2005.

[6] M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *Submitted to IEEE Trans. on Image Processing*.