

MOVING-WINDOW VARYING SIZE 3D TRANSFORM-BASED VIDEO DENOISING

Dmytro Rusanovskyy, Kostadin Dabov, and Karen Egiazarian

Institute of Signal Processing, Tampere University of Technology, Finland
 PO BOX 553, 33101 Tampere, Finland
 firstname.lastname@tut.fi

ABSTRACT

In this paper we consider the problem of suppressing additive noise in video data. We propose a transform-based video denoising method in sliding, local 3D variable-sized windows. For every spatial position in each frame we use a block-matching algorithm to collect highly correlated blocks from neighboring frames and form 3D arrays for all predefined window sizes by stacking the matched blocks. An optimal window size is then selected according to the ICI rule and a 3D unitary transform is applied to the selected 3D array. Hard-thresholding on its coefficients attenuates the noise and an inverse 3D transform reconstructs a local estimate of the noise-free signal in the array. The final estimate is a weighted average of the overlapping local ones. Our experiments show that the proposed algorithm outperforms all, known to the authors, video denoising methods, both in terms of objective criteria (L^2 distance) and visual quality.

1. INTRODUCTION

Despite the significant progress in video acquisition technologies, imperfect instruments, natural phenomena, transmission errors, and coding artifacts degrade the quality of video data by inducing noise. In many video applications the noise is modeled as additive white Gaussian noise (WGN). In this paper we consider the problem of suppressing additive WGN in video signals.

Recent research on the topic of video denoising ([1], [2], and [3]) has shown that methods operating in a transform domain and utilizing the temporal redundancy of video data provide a significantly superior denoising performance as compared to the spatial or spatial-temporal techniques.

The sliding-window 3D DCT video denoising method introduced in our earlier work [3] effectively attenuated noise in local 3D DCT domain. Let us recall its basic principle. For each processed location we formed highly correlated 3D arrays by applying block-matching and stacking the matched blocks together. Hard-thresholding of the 3D transform coefficients attenuated the noise and

inverse 3D DCT reconstructed estimates of the matched blocks. Similarly to current video coding standards, such as H.263, the block size was fixed in spatial domain to 8×8 pixels.

However, the noise-free signals within spatial blocks of fixed size, even as small as 8×8 , often contain details (e.g. sharp edges) which are not sparsely represented in transform domain. This deteriorates the performance of transform-based video processing techniques such as coding and denoising. As a possible solution of this problem for video coding, *variable block-size motion compensation* has been proposed as part of the novel standard H.264 [4]. In particular, blocks of size 16×16 down to 4×4 pixels are employed for motion prediction.

In this article, we extend the sliding-window 3D DCT video denoising algorithm [3] by adopting 3D windows with variable size in both spatial and temporal dimensions. We select an optimal 3D window size according to the Intersection of Confidence Intervals (ICI) rule [6] for each processed location. In this manner we adapt to the structures of the underlying, noise-free signal, and improve both the detail preservation and the noise suppression.

In Section 2, we develop our 3D transform-based video denoising algorithm with varying window-sizes. We present experimental results in Section 3 and, finally, give conclusions in Section 4.

2. VIDEO DENOISING IN 3D ADAPTIVE-WINDOW LOCAL TRANSFORM DOMAIN

Let us introduce the observation model and notation used throughout the paper. We consider a noisy observation $y(t) = x(t) + n(t)$, where $t \in V$ is a 3D coordinate that belongs to the spatial-temporal domain $V \subset Z^3$ of the video data, x is a noise-free video signal and $n(t)$ is WGN with variance σ_n^2 . Let us define a set of 3D window sizes, $\mathbf{S} = \{s \mid s \in \mathbb{N}^3\}$ ranged in ascending order so that an arbitrary window support $s \in \mathbf{S}$ embeds all smaller window supports. When we say that a block (2D

patch) is located at a 3D coordinate $t = \langle t_1, t_2, t_3 \rangle \in V$, we mean that the block is in the same temporal plane (frame), t_3 , and its upper-left element is positioned at spatial coordinate $\langle t_1, t_2 \rangle$.

For every processed location $t \in V$, we do the following steps. For each $s = \langle s', s'', s''' \rangle \in \mathbf{S}$, we apply the block-matching procedure in order to find s''' blocks of size $s' \times s''$ which are highly correlated to the reference one, located at t . The search is done within a spatial local neighborhood of size $N_s \times N_s$ among s''' frames centered about the temporal coordinate of the current location (see Figure 1). We form 3D arrays $B_{t,s \in \mathbf{S}}$ by stacking the matched blocks for each $s \in \mathbf{S}$. Then we select an optimal, according to the ICI rule, 3D window size $s_{opt,t} \in \mathbf{S}$. We filter locally the selected 3D array, $B_{t,s_{opt,t}}$, by hard-thresholding in a 3D transform domain. An inverse transform of the thresholded coefficients reconstructs a local estimate of the noise-free signal in $B_{t,s_{opt,t}}$,

$$\hat{B}_{t,s_{opt,t}} = F_{3D}^{-1} \left(T \left(F_{3D} \left(B_{t,s_{opt,t}} \right), \lambda_{thr} \right) \right), \quad (1)$$

where F_{3D} is a 3D unitary transform operator and T is a hard-thresholding operator based on the universal threshold from [5], $\lambda_{thr} \sigma_n \sqrt{2 \cdot \log \left(\left| B_{t,s_{opt,t}} \right| \right)}$, where $\left| B_{t,s_{opt,t}} \right|$ is the number of elements in $B_{t,s_{opt,t}}$. In order to reflect the relevancy of the reconstructed local estimates, we define a weight,

$$w_t = \frac{1}{N_{har}}, \quad (2)$$

where N_{har} is the number of non-zero transform coefficients after hard-thresholding. Observe that the total residual noise energy of $\hat{B}_{t,s_{opt,t}}$ is equal to $\sigma_n^2 N_{har}$. Thus, 3D arrays with sparser decompositions (i.e. smaller N_{har}) are awarded greater weights by (2).

After processing all locations $t \in V$, the reconstructed local estimates $\hat{B}_{t,s_{opt,t}}$, $t \in V$, in general, form an overcomplete representation of the video signal due to an overlap between them. In order to produce the final estimate of the noise-free video data, we aggregate the local estimates from (1) by a weighted average at the positions where they overlap. Thus, we compute \hat{x} as

$$\hat{x}(k) = \frac{\sum_{t \in V} w_t \hat{B}_{t,s_{opt,t}}(k)}{\sum_{t \in V} w_t \chi_t(k)}, \quad \forall k \in V,$$

where $\hat{B}_{t,s_{opt,t}}(k)$ is an estimate of $x(k)$ if its support

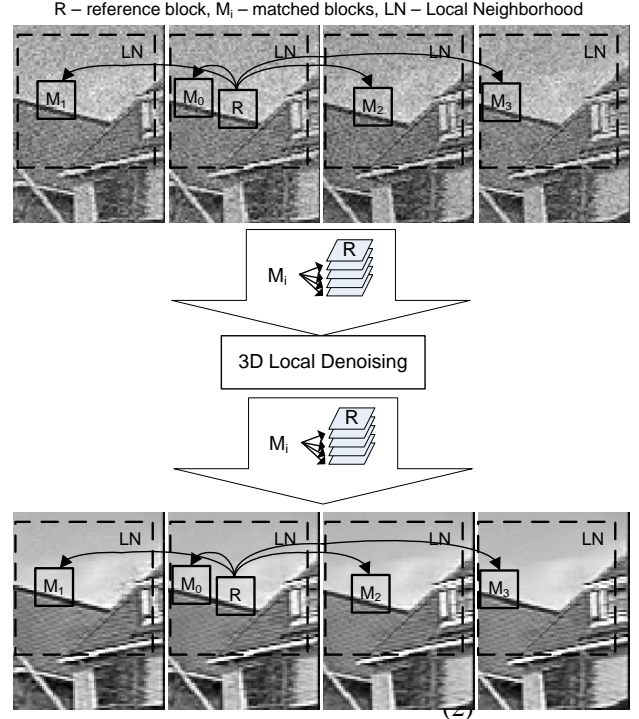


Figure 1. Example of searching for similar blocks with block-matching. The 3D array of size selected according to the ICI rule is then denoised in a local 3D transform domain.

overlaps k and zero otherwise, and where $\chi_t : V \rightarrow \{0,1\}$ is the characteristic function of the coordinates of the elements in $\hat{B}_{t,s_{opt,t}}$.

In the sub-sections to follow, we describe in detail the block-matching and the algorithm for optimal window-size selection.

2.1. Block-matching

Block-matching is employed to find blocks that exhibit high correlation to a given reference block. Because the accuracy of this operation is impaired by the presence of noise, we utilize a block-similarity measure which performs coarse initial denoising in local 2D transform domain prior to matching. Hence, we define a block-distance measure (inversely proportional to similarity) as

$$d(B_{t_1}, B_{t_2}) = \left\| T \left(F_{2D} \left(B_{t_1} \right), \lambda'_{thr} \right) - T \left(F_{2D} \left(B_{t_2} \right), \lambda'_{thr} \right) \right\|_2,$$

where F_{2D} is a 2D unitary transform operator and B_{t_1} and B_{t_2} are arbitrary blocks (2D patches) located at t_1 and t_2 , respectively. Thus, we apply block-matching in order to find the blocks with smallest d -distance to the currently-processed one.

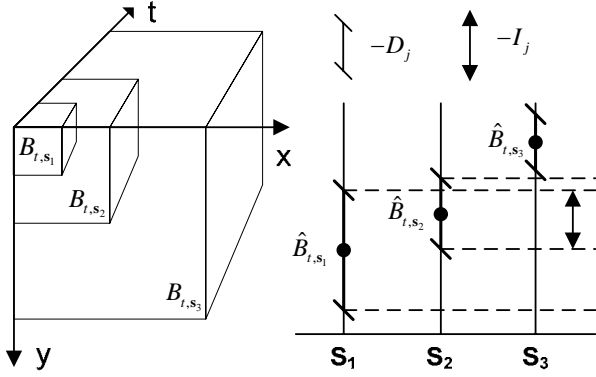


Figure 2. Selection of an optimal 3D window size according to the ICI rule.

2.2. ICI rule for optimal window-size selection

In order to adapt to the structures of the underlying signal, we use variable-sized 3D windows for denoising. The variable sizes help to avoid the unwanted presence of details which are not sparsely represented in transform domain (e.g. singularities and edges). Hence, by improving sparsity, we achieve better noise attenuation and improved detail preservation. We use the ICI rule [6] to determine an optimal 3D window-size for each processed location.

Recall that for every processed location $t \in V$, by block-matching, we obtain $|\mathbf{S}|$ 3D arrays $B_{t,s_j \in \mathbf{S}}$ whose sizes $s_j \in \mathbf{S}$ are the ordered elements of \mathbf{S} . The mean estimator applied on B_{t,s_j} gives

$$\tilde{B}_{t,s_j} = \text{mean}(B_{t,s_j}), \quad (3)$$

with variance

$$\sigma_{\tilde{B}_{t,s_j}}^2 = \frac{\sigma_n^2}{s_j' s_j'' s_j'''} \quad (4)$$

We use the variance to determine a confidence interval D_j for \tilde{B}_{t,s_j} . Thus, we define D_j as

$$D_j = \left[\tilde{B}_{t,s_j} - \lambda \cdot \sigma_{\tilde{B}_{t,s_j}}, \tilde{B}_{t,s_j} + \lambda \cdot \sigma_{\tilde{B}_{t,s_j}} \right], \quad (5)$$

where $\lambda > 0$ is a fixed parameter.

We apply the ICI rule for adaptive 3D window size selection as following (see Figure 2 for details).

1. Pre-compute the mean estimator's variances for each element of \mathbf{S} as given by (4).
2. For every processed location $t \in V$, compute \tilde{B}_{t,s_j} and the corresponding confidence intervals D_j as defined in Equations (3) and (5), respectively, for each $s_j \in \mathbf{S}$.

3. Check the intersection of the confidence intervals $I_j = \bigcap_{i=1}^j D_i$ for each s_j , where $j = 1 \dots |\mathbf{S}|$.
4. The largest index j for which I_j is non-empty determines the selected window size $s_{opt,t} = s_j$.

The adaptive window size is defined as the largest window size whose confidence interval intersects with the confidence intervals of all smaller window sizes.

As a result, the signal within the selected 3D array is characterized by a desired (controlled) level of homogeneity in both spatial and temporal dimensions. Therefore, a unitary transform can effectively decorrelate the true-signal energy in it.

3. RESULTS

We evaluate the performance of the proposed denoising algorithm for a few standard grayscale video signals: *Salesman*, *Tennis*, and *Flower*. In Table 1, we present the output ISNR results of: the proposed method, *Soft3D* [1], *WRSTF* [2], and *3DDCT* [3]. In Figure 3, we compare the PSNR-per-frame results of these methods for *Salesman*.

In our experiments we used the 2D DCT and the 3D DCT for the transform operators F_{2D} and F_{3D} , respectively. We have chosen DCT because of its good decorrelation ability and the availability of fast algorithms for lengths that are powers of 2.

We conducted all experiments with the following fixed settings. The set of predefined 3D window-sizes was $\mathbf{S} = \{\langle 4, 4, 8 \rangle, \langle 8, 8, 8 \rangle, \langle 16, 16, 8 \rangle\}$. The ICI rule's parameter was $\lambda = 1.5$. For block-matching, we used a spatial local neighborhood with length of the side $N_s = 15$.

In order to restrict complexity, we process the locations $t \in V$ in both spatial dimensions by sliding to every next processed location with a fixed step p which we denominate as "sliding step". Hence, it controls the overlap between adjacently processed blocks.

We did experiments for full-sliding ($p=1$) type of processing which is characterized by plenty of overlapping local estimates, thus a high level of overcompleteness. In spite of the high computational demand for such kind of processing, we show that the denoising results of this approach (labeled *3D-T1* in Table 1) are superior to the results of all other techniques.

We demonstrate the complexity scalability of our method in case of a larger sliding step, $p=4$ (see *3D-T4* in Table 1). It results in a 16-times decrease of the complexity as compared to the case of $p=1$.

We show a fragment of a noisy and denoised 11th frame of *Salesman* in Figure 3. The sharp details are well-preserved and also no visible artifacts are present in our estimate.

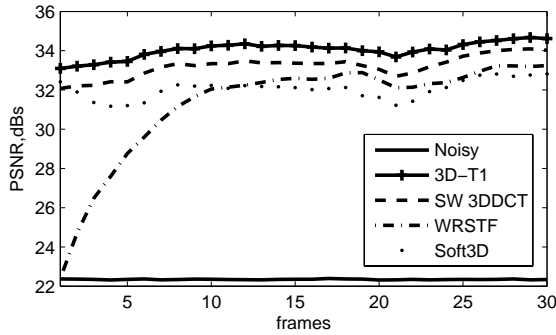


Figure 3. PSNR-per-frame comparative results for *Salesman* corrupted with additive WGN with $\sigma_n = 20$.

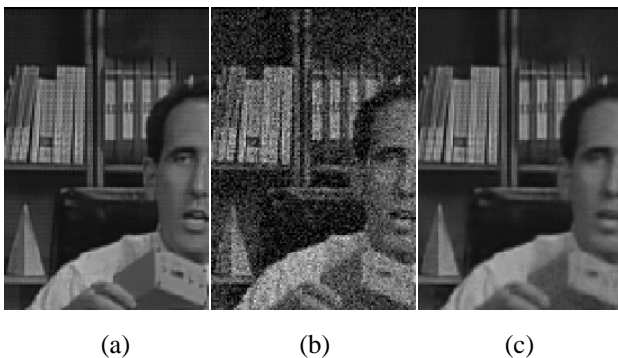


Figure 4. Fragment of the 11th frame of *Salesman*: (a) original, (b) noisy, $\sigma_n = 20$, PSNR 22.35 dB, (c) denoised by *3D-T1*, PSNR 34.27 dB.

4. CONCLUSIONS

Considering the sparsity of the transform-domain representations of 3D arrays as a key factor influencing the performance of our transform-based scheme, we do local filtering in moving, varying in size 3D windows. This results in good decorrelation of the noise-free signal in a local 3D unitary transform domain, hence the good detail preservation and effective noise suppression of our technique.

The results demonstrate that the proposed algorithm outperforms all other, known to the authors, video denoising methods, both in terms of PSNR and visual quality. Moreover, the method allows for effective computational scalability as a tradeoff between denoising performance and computational complexity.

5. REFERENCES

[1] W.I. Selesnick and K.Y. Li, "Video denoising using 2d and 3d dualtree complex wavelet transforms," in *Wavelet Applications in Signal and Image Processing, Proc. SPIE 5207*, San Diego, USA, August 2003.

[2] V. Zlokolica, A. Pizurica and W. Philips, "Wavelet Domain Noise-Robust Motion Estimation and Noise Estimation for Video Denoising," *First International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Scottsdale, Arizona, USA, January 23-25, 2005.

[3] D. Rusanovskyy and K. Egiazarian, "Video Denoising Algorithm in Sliding 3D DCT Domain," in *Proc. of Advanced Concepts for Intelligent Video Systems, ACIVS 2005*, Antwerp, Belgium, September 20-23, 2005.

[4] JVT of ITU-T and ISO/IEC JTC 1, "Draft ITU-T Recom. and Final Draft Int. Standard of Joint Video Specific. (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)", document JVT-G050r1, May 2003; and Fidelity Range Extensions documents JVT-L047 (non-integrated form) and JVT-L050 (integrated form), July 2004.

[5] D.L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. on Information Theory*, vol. **41**, no. 3, pp. 613-627, May 1995.

[6] V. Katkovnik, "A new method for varying adaptive bandwidth selection," *IEEE Trans. on Signal Processing*, vol. **47**, pp. 2567-2571, September 1999.

Table 1. Results in output ISNR (dB).

Video sequence	Noise σ_n /PSNR	ISNR (dB)				
		<i>Soft3D</i> [1]	<i>WRSTF</i> [2]	<i>3DDCT</i> [3]	<i>3D-T1</i>	<i>3D-T4</i>
<i>Salesman</i>	15/24.64	8.57	9.19	10.11	10.94	10.03
	20/23.12	9.65	10.05	10.94	11.68	10.82
<i>Tennis</i>	15/26.02	5.23	5.49	6.17	6.47	5.95
	20/22.15	6.43	6.53	7.07	7.36	6.91
<i>Flower</i>	15/25.33	2.8	3.31	4.04	4.80	4.27
	20/23.25	2.57	3.95	4.62	5.14	4.48