Pre-print.

To appear in Proc. of the 15-th European Signal Processing Conference (EUSIPCO), Poznań, Poland, September 3-7, 2007.

VIDEO DENOISING BY SPARSE 3D TRANSFORM-DOMAIN COLLABORATIVE FILTERING

Kostadin Dabov, Alessandro Foi, and Karen Egiazarian

Institute of Signal Processing, Tampere University of Technology
P.O. Box 553, 33101 Tampere, Finland
firstname.lastname@tut.fi

ABSTRACT

We propose an effective video denoising method based on highly sparse signal representation in local 3D transform domain. A noisy video is processed in blockwise manner and for each processed block we form a 3D data array that we call "group" by stacking together blocks found similar to the currently processed one. This grouping is realized as a spatio-temporal predictive-search block-matching, similar to techniques used for motion estimation. Each formed 3D group is filtered by a 3D transform-domain shrinkage (hard-thresholding and Wiener filtering), the result of which are estimates of all grouped blocks. This filtering—that we term "collaborative filtering"—exploits the correlation between grouped blocks and the corresponding highly sparse representation of the true signal in the transform domain. Since, in general, the obtained block estimates are mutually overlapping, we aggregate them by a weighted average in order to form a non-redundant estimate of the video. Significant improvement of this approach is achieved by using a two-step algorithm where an intermediate estimate is produced by grouping and collaborative hard-thresholding and then used both for improving the grouping and for applying collaborative empirical Wiener filtering. We develop an efficient realization of this video denoising algorithm. The experimental results show that at reasonable computational cost it achieves state-of-the-art denoising performance in terms of both peak signal-to-noise ratio and subjective visual quality.

1. INTRODUCTION

Many video denoising methods have been proposed in the last few years. Prominent examples of the current developments in the field are the wavelet based techniques [1, 2, 3, 4, 5]. These methods typically utilize both the sparsity and the statistical properties of a multiresolution representation as well as the inherent correlations between frames in temporal dimension. A recent denoising strategy, the non-local spatial estimation [6], has also been adapted to video denoising [7]. In this approach, similarity between 2D patches is used to determine the weights in a weighted averaging between the central pixels of these patches. For image denoising, the similarity is measured for all patches in a 2D local neighborhood centered at the currently processed coordinate. For video denoising, a 3D such neighborhood is used. The effectiveness of this method depends on the presence of many similar true-signal blocks.

Based on the same assumption as the one used in the non-local estimation, i.e. that there exist mutually similar blocks in natural images, in [8] we proposed an image denoising method. There, for each processed block, we perform two special procedures — *grouping* and *collaborative filtering*. Grouping finds mutually similar 2D blocks and then stacks them together in a 3D array that we call *group*. The benefit of grouping highly similar signal fragments together is the increased correlation of the true signal in the formed 3D array. Collaborative filtering takes advantage of this increased correlation to effectively suppress the noise and produces estimates of each of the grouped blocks. We showed [8] that this approach is very effective for image denoising.

In this paper, we apply the concepts of grouping and collaborative filtering to video denoising. Grouping is performed by a specially developed predictive-search block-matching technique that significantly reduces the computational cost of the search for similar blocks. We develop a two-step video-denoising algorithm where the predictive-search block-matching is combined with collaborative hard-thresholding in the first step and with collaborative Wiener filtering in the second step. At a reasonable computational cost, this algorithm achieves state-of-the-art denoising results in terms of both PSNR and visual quality. This work generalizes the denoising approach from [8] and improves on the video denoising algorithm proposed in [9].

2. GROUPING AND COLLABORATIVE FILTERING FOR VIDEO DENOISING

The concepts of grouping and collaborative filtering are both extensively studied in [8]. Therefore, in this section we only give a general overview in the context of video denoising. A noisy video signal is processed in block-wise manner (processed blocks can overlap), where the currently processed block is denominated *reference block*. For each reference block, grouping is performed followed by collaborative filtering.

Analogously to [8], we realize grouping by block-matching, a procedure that tests the similarity between the reference block and ones that belong to a predefined search neighborhood. The similarity is typically computed as the inverse of some distance (dissimilarity) measure. The distance that we adopt in the sequel is the ℓ^2-norm of the difference between two blocks. Given the nature of video, the search neighborhood is a 3D domain that spans both the temporal and the two spatial dimensions. In this work, we propose to search for similar blocks by *predictive-search block-matching*. The peculiarity of this technique, fully explained in Section 4, is the adoption of data-adaptive spatio-temporal 3D search neighborhoods. They are adaptive to similarities between and within the frames, and thus to motion in the video. It allows for a significant complexity reduction as compared with full-search in non-adaptive neighborhoods.

In [8] we demonstrated that transform-domain shrinkage can be utilized as an effective realization of collaborative filtering. It comprises three steps; first, a 3D transform is applied on a group to produce a highly sparse representation of the true signal in it; second, shrinkage (e.g., hard thresholding or Wiener filtering) is performed on the transform coefficients; and third, an inverse 3D transform produces estimates of all grouped blocks. By exploiting the similarity

This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program [2006 - 2011]).

among grouped blocks, the transform produces a sparse representation of the true signal in the transform domain. This enables the subsequent shrinkage to efficiently attenuate the noise and at the same time to preserve the most significant portion of the true-signal spectrum.

After performing grouping and collaborative filtering for each reference block, a collection of overlapping blockwise estimates is obtained. This collection forms a redundant estimate of the true signal. In order to form a non-redundant estimate, the blockwise estimates need to be aggregated. As in [8], we propose aggregation by weighted averaging where the weights are inversely proportional to the squared ℓ^2-norm of the shrunk groups' spectra and thus loosely reciprocal to the total variance of each filtered group.

Using the above procedures, we develop a two-step video-denoising algorithm whose structure is analogous to that of the image-denoising algorithm [8]. The first step produces a basic (intermediate) estimate of the video signal by applying the proposed denoising scheme using grouping and collaborative hard-thresholding. The second step uses the basic estimate to improve the denoising in the following two aspects. First, grouping is performed within the basic estimate rather than within the noisy video, and second, the hard-thresholding is replaced by empirical Wiener filtering that uses the spectra of groups from the basic estimate.

3. ALGORITHM

We consider an observed noisy video $z(x) = y(x) + \eta(x)$, where y is the true video signal, $\eta(\cdot) \sim \mathcal{N}(0, \sigma^2)$ is an i.i.d. Gaussian noise sample and $x = (x_1, x_2, t) \in X$ are coordinates in the spatio-temporal 3D domain $X \subset \mathbb{Z}^3$. The first two components $(x_1, x_2) \in \mathbb{Z}^2$ are the spatial coordinates and the third one, $t \in \mathbb{Z}$, is the time (frame) index. The variance σ^2 is assumed a priori known. The proposed two-step denoising algorithm is presented in the right column of this page. It is also illustrated in Figure 1.

4. GROUPING BY PREDICTIVE-SEARCH BLOCK-MATCHING

A straightforward approach is to use a fixed-size 3D search neighborhood for the grouping by block-matching. However, capturing blocks of a moving object across many frames requires large spatial dimensions of such search neighborhood. On the one hand, using large sizes imposes a rather high complexity burden, and on the other hand, using small ones results in unsatisfactory grouping and poor denoising results.

In order to efficiently capture blocks that are part of objects which move across subsequent frames, we propose to use predictive-search block-matching, an inductive procedure that finds similar (matching) blocks by searching in a data-adaptive spatio-temporal subdomain of the video sequence. For a given reference block located at $x = (x_1, x_2, t_0)$, when using a temporal window of $2N_{FR}+1$ frames, the predictive-search block-matching comprises the following steps.

- Starting with frame t_0, an exhaustive-search block-matching is performed in a nonadaptive $N_S \times N_S$ neighborhood centered about (x_1, x_2). The result are the spatial locations of the N_B blocks (within this neighborhood) which exhibit highest similarity to the reference one. These locations are collected in the set $S^{t_0} \subset \mathbb{Z}^3$.
- The *predictive search* in frame $t_0 + k$, $0 < |k| \leq N_{FR}$, is defined inductively based on the matching results from the previously processed frame $t_0 + k - sign(k)$, i.e. from the preceding frame for $k > 0$ or from the subsequent frame for $k < 0$. This search for similar blocks takes place within the union of $N_{PR} \times N_{PR}$ neighborhoods centered at the spatial coordinates of the previously found locations $x \in S^{t_0+k-sign(k)}$. That is, these locations predict

V-BM3D video denoising algorithm

Step 1. Obtain a basic estimate using grouping and collaborative hard-thresholding.

1.1. For each coordinate $x \in X_R$ do:

(a) $S_x = PS\text{-}BM(Z_x)$,

(b) $\hat{\mathbf{Y}}_{S_x} = T_{3D}^{-1}\left(HARD\text{-}THR\left(T_{3D}\left(\mathbf{Z}_{S_x}\right), \lambda_{3D}\sigma\right)\right)$, where $\hat{\mathbf{Y}}_{S_x}$ is a group of blockwise estimates $\hat{Y}_{x'}^{x}$, $\forall x' \in S_x$.

1.2. Produce the basic estimate $\hat{y}^{basic}$ by aggregation of the blockwise estimates $\hat{Y}_{x'}^{x}$, $\forall x \in X_R$ and $\forall x' \in S_x$ using weighted averaging with $weight\left(\hat{Y}_{x'}^{x}\right) = \frac{1}{\sigma^2 N_{har}(x)} W_{2D}$.

Step 2. Obtain the final estimate by grouping within the basic estimate and collaborative Wiener filtering that uses the spectra of the corresponding groups from the basic estimate.

2.1. For each coordinate $x \in X_R$ do:

(a) $S_x = PS\text{-}BM\left(\hat{Y}_{x}^{basic}\right)$,

(b) $\hat{\mathbf{Y}}_{S_x} = T_{3D}^{-1}\left(T_{3D}\left(\mathbf{Z}_{S_x}\right) \frac{\left[T_{3D}\left(\hat{\mathbf{Y}}_{S_x}^{basic}\right)\right]^2}{\left[T_{3D}\left(\hat{\mathbf{Y}}_{S_x}^{basic}\right)\right]^2 + \sigma^2}\right)$.

2.2. Produce the final estimate $\hat{y}^{final}$ by aggregation of $\hat{Y}_{x'}^{x}$ $\forall x \in X_R$ and $\forall x' \in S_x$ using weighted averaging with $weight\left(\hat{Y}_{x'}^{x}\right) = \sigma^{-2} \left\| \frac{\left[T_{3D}\left(\hat{\mathbf{Y}}_{S_x}^{basic}\right)\right]^2}{\left[T_{3D}\left(\hat{\mathbf{Y}}_{S_x}^{basic}\right)\right]^2 + \sigma^2} \right\|_2^{-2} W_{2D}$, where $\|\cdot\|_2$ denotes ℓ^2-norm.

Notation:

- $X_R \subset X$ is a set that contains the coordinates of the processed reference blocks. We build it by taking each N_{step} element of X along both spatial dimensions, hence $|X_R| \approx \frac{|X|}{N_{step}^2}$.
- Z_x denotes a block of size $N_1 \times N_1$ in z, whose upper-left corner is at x. Similar notation is used for $\hat{Y}_{x'}^{x}$ and $\hat{Y}_{x}^{basic}$; the former is an estimate for the block located at x', obtained while processing reference block Z_x and the latter is a block located at x extracted from the basic estimate y^{basic}.
- $S_x = PS\text{-}BM(Z_x)$ performs predictive-search block-matching (Section 4) using Z_x as a reference block, the result of which is the set S_x containing the coordinates of the matched blocks. For Step 2, the search is performed in the basic estimate instead of in the noisy video.
- $\mathbf{Z}_{S_x}$ denotes a group (i.e. a 3D array) formed by stacking together the blocks $Z_{x \in S_x}$; the same notation is used for $\hat{\mathbf{Y}}_{S_x}^{x}$ and $\hat{\mathbf{Y}}_{S_x}^{basic}$. The size of these groups is $N_1 \times N_1 \times |S_x|$.
- $T_{3D}\left(\mathbf{Z}_{S_x}\right)$ is the spectrum of $\mathbf{Z}_{S_x}$ using a 3D linear transform T_{3D} which should have a DC basis element (e.g. the 3D-DCT, the 3D-DFT, etc.).
- $HARD\text{-}THR(T_{3D}\left(\mathbf{Z}_{S_x}\right), \lambda_{3D}\sigma)$ applies hard-thresholding on the transform coefficients (except for the DC) using threshold $\lambda_{3D}\sigma$ where λ_{3D} is a fixed threshold parameter.
- $N_{har}(x)$ is the number of nonzero coefficient retained after hard-thresholding $T_{3D}\left(\mathbf{Z}_{S_x}\right)$; since the DC is always preserved, $N_{har}(x) > 0$, ensuring division by zero never occurs in Step 1.1.a.
- W_{2D} is a 2D Kaiser window of size $N_1 \times N_1$.

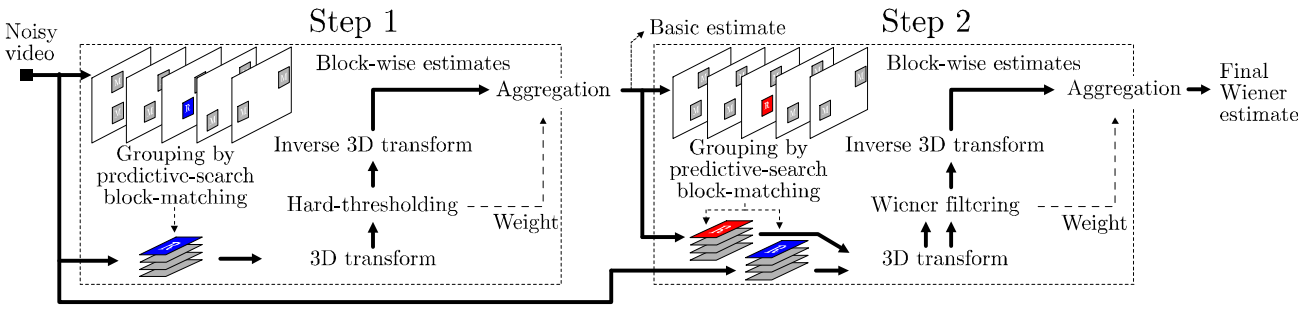

Figure 1: Flowchart of the proposed BM3D video denoising method. The operation enclosed by dashed lines are repeated for each reference block. Grouping is illustrated by showing a reference block marked with 'R' and the matched ones in a temporal window of 5 frames ($N_{FR} = 2$).

where similar blocks are likely to be present in the current frame (i.e. frame $t_0 + k$) and thus one can afford to have $N_{PR} < N_S$. The result for the current frame are the N_B locations of the blocks that exhibit highest similarity to the reference one; they are collected in the set S^{t_0+k}.

After performing the predictive-search block-matching for all of the frames $t_0 + k$ for $k = -N_{FR}, \ldots, N_{FR}$, we form a single set $S_x \subset \mathbb{Z}^3$ that contains at most N_2 of all $x' \in \bigcup_{k=-N_{FR}}^{N_{FR}} S^{t_0+k}$ that have the smallest corresponding block-distances to the reference block, which distances should also be smaller than a predefined threshold, τ_{match}. A group is later formed by stacking together blocks located at $x' \in S_x$. The exact ordering of the blocks within the 3D groups is not important, as shown in [8]. In the worst case, no matching blocks are found and then the group will contain only one block — the reference one — since its distance to itself is zero and therefore x will always be included in S_x.

Except for the frame t_0 in the procedure presented above, the spatial search neighborhoods are data-adaptive as they depend on previously matched locations. This adaptivity can be interpreted as following the motion of objects across frames. It is worth noting that similar approach has already been used for motion estimation [10] and also for fractal based image coding [11].

5. RESULTS

We present experimental results obtained with the proposed V-BM3D algorithm. A Matlab implementation of the V-BM3D that can reproduce these results is publicly available at `http://www.cs.tut.fi/~foi/GCF-BM3D`. There, one can find original and processed test sequences, details of which can be seen in Table 1.

The same algorithm parameters were used in all experiments. Here we give the most essential ones, as the rest can be seen in the provided Matlab script. The temporal window used 9 frames, i.e. $N_{FR} = 4$. The predictive-search block-matching used $N_S = 7$, $N_{PR} = 5$, and $N_B = 2$; the maximum number of matched blocks was $N_2 = 8$, and the threshold $\lambda_{3D} = 2.7$. Some of the parameters differed for the two steps; i.e., for Step 1, $N_1 = 8$, $N_{step} = 6$, and for Step 2, $N_1 = 7$, $N_{step} = 4$. The transforms were the same as in [8]: for Step 1, T_{3D} is a separable composition of a 1D biorthogonal wavelet full-dyadic decomposition in both spatial dimensions and a 1D Haar wavelet full-dyadic decomposition in the third (temporal) dimension; for Step 2, T_{3D} uses the 2D DCT in spatial domain and the same Haar decomposition in the temporal one. To increase the number of non-overlapping blocks in the groups and hence have more uncorrelated noise in them, we slightly modified the used distance measure. The modification was a subtraction of a small value d_s ($d_s = 3$ for Step 1 and $d_s = 7$ for Step 2) from the distance computed for blocks that are at the spatial coordinate of the reference one but in different frames.

In Table 1 we present the PSNR (dB) results of the proposed algorithm for a few sequences; there, the PSNR was measured globally on each whole sequence. In Figure 2, we compare our method with the 3DWTF [5] and the WRSTF [3], which are among the state-of-the-art in video denoising. For this comparison, we applied our method on noisy sequences and compared with the ones denoised by the other two methods. These sequences had been made publicly available by Dr. V. Zlokolica at `http://telin.ugent.be/~vzlokoli/PHD`, for which we are thankful. We note that the pixel intensities of the input noisy videos are quantized to integers in the range $[0, 255]$, unlike in the case of the results in Table 1. In Figure 2 one can observe that the proposed V-BM3D produces significantly higher PSNR than the other two methods for each frame of the three considered sequences, with a difference well higher than 1 dB for most of the frames. Moreover, this was achieved at similar execution times as compared with the WRSTF; i.e., the proposed V-BM3D (implemented as a Matlab MEX-function) filters a CIF (288×352) frame for 0.7 seconds on a 1.8 GHz Intel Core Solo machine and the WRSTF was reported [3] to do the same for 0.86 seconds on an Athlon64 (4000+) 2.4 GHz machine. Figure 3 gives a visual comparison for a fragment of the 77th frame of *Tennis* denoised with each of the considered techniques. The proposed method shows superior preservation of fine image details and at the same time it introduces significantly less artifacts.

6. DISCUSSION

Let us compare the proposed predictive-search block-matching with the motion estimation methods based on block-matching. Indeed, the predictive-search block-matching proposed in this work can be viewed as a sophisticated motion estimation which is not restricted to only one matched block per frame. That is, N_B blocks per frame can be used in the proposed grouping scheme. This can be beneficial in situations when there are only very few (or none) similar blocks along the temporal dimension, e.g. in the case of frame change. In that case, mutually similar blocks at different spatial locations within the same frame are exploited when forming groups and hence better sparsity is achieved by applying a 3D transform. This can be particularly effective when grouping blocks that are parts of, e.g., edges, textures, and uniform regions.

The second step of the proposed method is very impor-

	Video name:	*Salesm.*	*Tennis*	*Fl.Gard.*	*Miss Am.*	*Coastg.*	*Foreman*	*Bus*	*Bicycle*
	Frame size:	288×352	240×352	240×352	288×360	144×176	288×352	288×352	576×720
σ/PSNR	Frames:	50	150	150	150	300	300	150	30
5 / 34.15		40.44	38.47	36.49	41.58	38.25	39.77	37.55	40.89
10 / 28.13		37.21	34.68	32.11	39.61	34.78	36.46	33.32	37.62
15 / 24.61		35.44	32.63	29.81	38.64	33.00	34.64	31.05	35.67
20 / 22.11		34.04	31.20	28.24	37.85	31.71	33.30	29.57	34.18
25 / 20.17		32.79	30.11	27.00	37.10	30.62	32.19	28.48	32.90
30 / 18.59		31.68	29.22	25.89	36.41	29.68	31.27	27.59	31.77
35 / 17.25		30.72	28.56	25.16	35.87	28.92	30.56	26.91	30.85

Table 1: Output PSNR (dB) of the proposed V-BM3D algorithm for a few image sequences; the noise is i.i.d. Gaussian with variance σ^2 and zero mean. The PSNR was computed globally on each whole sequence.

σ	PSNR after Step 1	PSNR after Step 2
10	35.44	36.46
15	33.59	34.64
20	32.12	33.30
25	30.86	32.19

Table 2: PSNR improvement after applying Step 2 of our algorithm on the *Foreman* test sequence.

tant for the effectiveness of the overall approach. This is due to the improved grouping and the improved shrinkage by empirical Wiener filtering both of which are made possible by utilizing the basic (intermediate) estimate. In Table 2 one can compare the PSNR corresponding to both the basic estimate and the final one; the improvement is substantial, exceeding 1 dB in all cases shown there.

Since our approach uses the same assumptions that are used for the non-local estimation denoising, it is worth comparing the two approaches. The non-local means uses weighted averaging to obtain the final pixel estimate, where the weights depend on the similarity between the 2D patches (e.g. blocks) centered at the averaged pixels and the patch centered at the estimated pixel. In order to achieve good performance, this approach needs to capture plenty of very similar (in the ideal case, identical) patches. In the proposed method, we use a rather different approach where instead of a simple weighted average we use a complete decorrelating linear transform. The higher-order terms of the transform can approximate also variations between the spectral components of the grouped blocks, enabling a good filtering also for relatively dissimilar blocks. The subsequent shrinkage allows to take advantage of the sparsity by preserving the high-magnitude coefficients and truncating the ones with small magnitudes that are mostly due to noise.

The weights in the adopted aggregation (Steps 1.2 and 2.2 of the algorithm) favour blockwise estimates coming from sparsely represented groups. Such groups have few nonzero coefficients after hard-thresholding (Step 1.1.b) and few Wiener attenuation coefficients close to unity (Step 2.1.b).

Computational scalability of the V-BM3D can be achieved as in the image denoising counterpart of the algorithm [8] by varying certain parameters. The most important parameters that allow for such trade-off between denoising quality and complexity are the sliding step N_{step} and the block-matching parameters N_S, N_{PR}, and N_B.

7. CONCLUSIONS

In this work, we proposed a video denoising method that is both computationally efficient and achieves state-of-the-art results in terms of both PSNR and visual quality. These results are consistent with the ones already obtained by the image denoising counterpart [8] of the approach. We are currently working on extensions of the proposed method. A detailed analysis of its complexity and implementation issues as well as its application to color-video denoising will be reported in a forthcoming full-length publication.

REFERENCES

[1] E. Balster, Y. Zheng, and R. Ewing, "Combined spatial and temporal domain wavelet shrinkage algorithm for video denoising," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 16, no. 2, pp. 220–230, February 2006.

[2] F. Jin, P. Fieguth, and L. Winger, "Wavelet video denoising with regularized multiresolution motion estimation," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. Article ID 72 705, 11 pages, 2006.

[3] V. Zlokolica, A. Pizurica, and W. Philips, "Wavelet-domain video denoising based on reliability measures," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 16, no. 8, pp. 993–1007, August 2006.

[4] S. M. M. Rahman, M. O. Ahmad, and M. N. S. Swamy, "Video denoising based on inter-frame statistical modeling of wavelet coefficients," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 17, no. 2, pp. 187–198, February 2007.

[5] I. Selesnick and K. Li, "Video denoising using 2D and 3D dual-tree complex wavelet transforms," in *Proc. Wavelet Applicat. Signal Image Process. X, SPIE*, San Diego, USA, August 2003.

[6] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multisc. Model. Simulat.*, vol. 4, no. 2, pp. 490–530, 2005.

[7] A. Buades, B. Coll, and J. Morel, "Denoising image sequences does not require motion estimation," in *Proc. IEEE Conf. Adv. Video Signal Based Surveil., AVSS*, Palma de Mallorca, Spain, September 2005, pp. 70–74.

[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, August 2007, to appear in.

[9] D. Rusanovskyy, K. Dabov, and K. Egiazarian, "Moving-window varying size 3D transform-based video denoising," in *Proc. Int. Workshop on Video Process. and Quality Metrics*, Scottsdale, Arizona, USA, January 2006.

[10] C.-L. Fang, W.-Y. Chen, Y.-C. Liu, and T.-H. Tsai, "A new adaptive return prediction search algorithm for block matching," in *Proc. IEEE Pacific Rim Conference on Multimedia*, vol. 2532/2002, Hsinchu, Taiwan, December 2002, pp. 120–126.

[11] C.-C. Wan and C.-H. Hsieh, "An efficient fractal image-coding method using interblock correlation search," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 11, no. 2, pp. 257–261, February 2001.

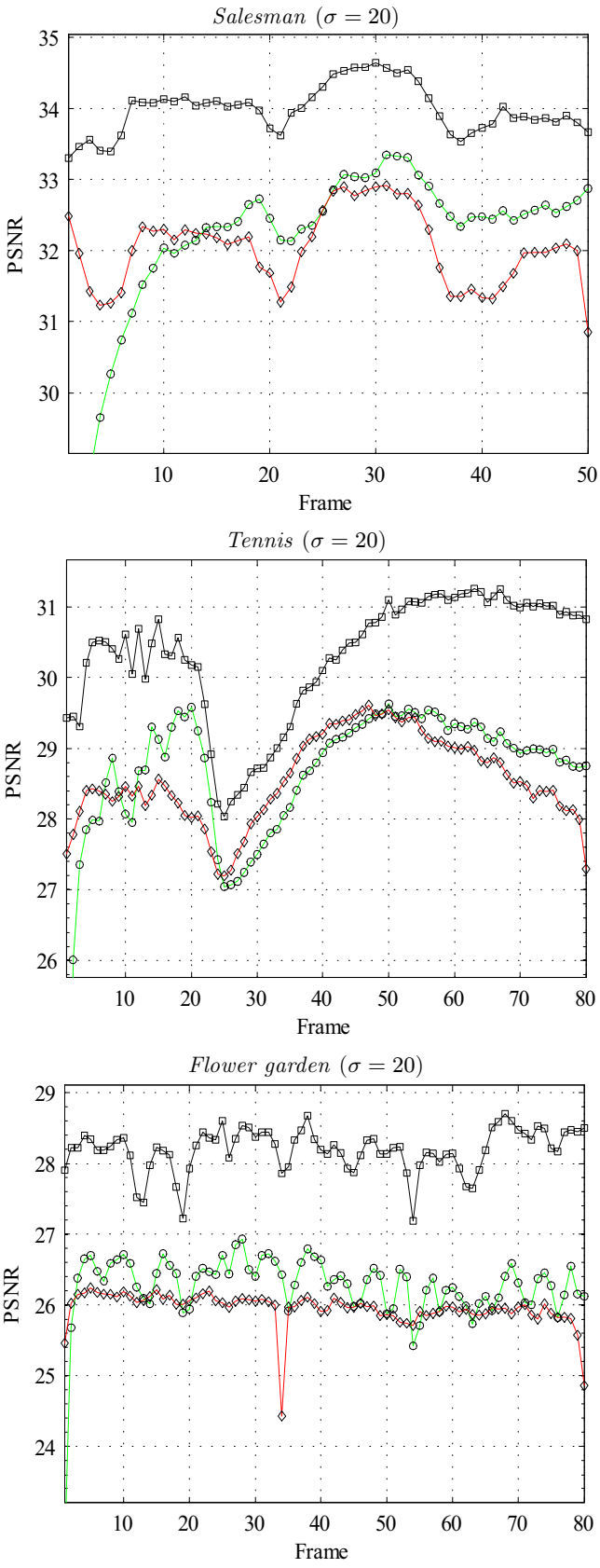

Figure 2: Per-frame PSNR comparison of the proposed V-BM3D ('□' marker) with the WRSTF [3] ('○' marker) and the 3DWTF [5] ('◇' marker).

3DWTF [5] (PSNR 28.12 dB)

WRSTF [3] (PSNR 28.83 dB)

Proposed V-BM3D (PSNR 30.93 dB)

Figure 3: Fragment of the 77th frame of *Tennis* denoised by the 3DWTF, the WRSTF, and the proposed V-BM3D; the noise had $\sigma = 20$. The output PSNR (for this frame only) is given in parentheses for each of the methods.