Video Signal-Dependent Noise Estimation via Inter-Frame Prediction

Yanhao Li, Marina Gardella, Quentin Bammey, Tina Nikoukhah, Rafael Grompone von Gioi, Miguel Colom, Jean-Michel Morel

VIDEO SIGNAL-DEPENDENT NOISE ESTIMATION VIA INTER-FRAME PREDICTION

Yanhao Li *Marina Gardella* *Quentin Bammey* *Tina Nikoukhah*
Rafael Grompone von Gioi *Miguel Colom* *Jean-Michel Morel*

Centre Borelli, ENS Paris-Saclay, Université Paris-Saclay, CNRS, France

ABSTRACT

We propose a block-based signal-dependent noise estimation method on videos, that leverages inter-frame redundancy to separate noise from signal. Block matching is applied to find block pairs between two consecutive frames with similar signal. Then Ponomarenko's method is extended by sorting pairs by their low-frequency energy and estimating noise in the high frequencies. Experiments on three datasets show that this method improves on the state of the art.

***Index Terms*—** Noise estimation, image processing, video processing, noise level function

1. INTRODUCTION

Noise estimation is a key preliminary step for various applications in image and video processing. An accurate noise level estimate can significantly boost the performance of downstream applications such as video denoising [1], forgery detection [2], camera identification, camera characterization and video quality assessment. Most noise estimation methods focus on single images [3]–[10]. These methods can be applied on each frame for video noise estimation but video temporal redundancy is not used. To this aim, we propose an extension of Ponomarenko's method [11] to videos. Since two consecutive frames mostly contain the same signal up to a local motion, noise can be separated by eliminating the underlying scene content within a flexible frame-to-frame difference.

2. RELATED WORK

A significant part of the literature on noise estimation focuses on single images, generally done by finding homogeneous regions where noise dominates. Noise in these regions is estimated in the spatial or frequency domain.

There are several approaches to identify homogeneous regions. Tai et al. [3] use a Sobel operator to discard edges, then apply a Laplacian operator to estimate noise. In [4], a hybrid discrete wavelet transform is used for edge-region removal. Colom et al. [5] select a small percentile of the image's blocks with the lowest standard deviations in the high frequencies. The bias induced by this selection criteria is then corrected. Instead, in [6], [7] a small percentile of blocks with the lowest low-frequency variances are selected. Noise is then estimated on the high-frequencies. Mohan et al. [8] first perform intra-image patch matching, then estimate noise in the discrete cosine transform (DCT) domain. Other methods use Principal Component Analysis (PCA) to find homogeneous patches. Pyatykh et al. [9] select patches with similar structure and small variance using PCA, then estimate noise from the smallest eigenvalues. Similarly, Liu et al. [10] present a PCA-based method to select low-rank patches with the smallest high frequency energy based on their gradients.

The availability of numerous samples in video can improve estimation. In [12], the noise level function (NLF) is estimated by selecting and clustering homogeneous frame regions. Inter-frame analysis then stabilizes temporal noise variations. Buades et al. [1] extend [7] to estimate an NLF using simultaneously all video frames. Temporal information is further employed to improve accuracy and robustness. In [13], the ideas presented in [14] are extended to the differential image obtained from two consecutive frames. They construct a homogeneity measure using high-pass operators along several directions to select the most homogeneous blocks for final estimation. Motion estimation is used to better handle inter-frame scene changes. Yin et al. [15] estimate noise on the residual of the motion estimation on half the matched blocks, to avoid overestimation due to mismatches. Xiao et al. [16] present an algorithm to suppress video motion by inter-frame block matching, and estimate noise level within the inter-frame difference with PCA.

Several approaches exploit jointly the spatial and temporal domains. Ghazal et al. [17] sort 3D cubes by their response to directional Laplacian operators. The homogeneous ones are used to estimate the noise level in each direction. The mean of these estimations yields a final estimate. An improved version of this method [18] uses instead Laplacian of Gaussian operators and a median estimator. Zlokolica et al. [19] employ spatio-temporal gradients of image sequence content with wavelet transform analysis to determine the noise variation. Spatio-temporal gradients are also used in [20] to select homogeneous cuboids, followed by adaptive noise estimation. Izadi et al. [21] address inter-frame correlated noise and compute the noise level by a combination of spatial and temporal variance estimations.

This work was supported by a grant from Région Île-de-France.

3. PROPOSED METHOD

Ponomarenko's method [6], [7], [11] was designed to estimate an NLF from a single image. We extend this method to leverage the temporal redundancy for video noise estimation. Like in the Ponomarenko method, the signal is filtered out with the low frequencies. Noise is then estimated in the high frequencies. The key difference is that instead of using the raw blocks in a single frame, we compute the residual of matched blocks (or difference blocks) between two consecutive frames.

The proposed algorithm works independently on color channels, estimating one NLF for each one. First, each frame is divided into $w \times w$ blocks and block matching is performed in two consecutive frames. The matched pairs are then classified into bins according to their intensities. The residual between each block pair is computed; the DCT is then applied to calculate the frequency coefficients of each difference block. Finally, in a similar way as in [6], [7], we select the difference blocks with the lowest energy at low frequencies and estimate the noise in the high frequencies of those blocks.

The signal-dependent noise in raw images can be modelled as Poisson-Gaussian and can be approximated by a signal-dependent Gaussian noise; its variance is an affine function of the signal intensity [11]. After undergoing the in-camera processing pipeline, the relation between the *noise variance* and the *expectation of intensity* becomes non-linear, but the noise is still signal-dependent. We denote a noisy pixel by $\tilde{\mathrm{I}} = \mathrm{I} + \mathrm{n}$ where I is the true signal and n the noise. The NLF g describing the relation between the signal intensity and the noise variance is $g(\mathrm{I}) \triangleq \mathrm{Var}(\tilde{\mathrm{I}}) = \mathrm{Var}(\mathrm{n})$.

Block matching Each frame is divided into overlapping blocks of size $w \times w$ with stride 1. Let $\{\tilde{\mathbf{U}}_k^t : k = 1, \dots, K\}$ denote the set of blocks extracted from frame t. The purpose of block matching is to find, for each block $\tilde{\mathbf{U}}_k^t$ in frame t, a corresponding block $\tilde{\mathbf{U}}_{k'}^{t+1}$ in frame $t+1$ having similar signal, so that the noise can be estimated from their difference without interference of the scene. Indeed, if $\tilde{\mathbf{U}}_k^t = \mathbf{U}_k^t + \mathbf{n}_k^t$, where $\mathbf{U}_k^t$ is the true signal and $\mathbf{n}_k^t$ is the noise, then:

$$\tilde{\mathbf{U}}_k^t - \tilde{\mathbf{U}}_{k'}^{t+1} = (\mathbf{U}_k^t - \mathbf{U}_{k'}^{t+1}) + (\mathbf{n}_k^t - \mathbf{n}_{k'}^{t+1}) \approx \mathbf{n}_k^t - \mathbf{n}_{k'}^{t+1}.$$

The usual way to determine k' is to search the block of frame $t+1$ (within a certain range) that minimizes the similarity distance. Here we use two metrics for the similarity distance: (a) the sum of absolute differences (SAD) and (b) the sum of gradient differences (SGD). Then,

$$k' = \underset{c' \in \text{candidates}}{\arg\min} \ \mathrm{Dist}(\tilde{\mathbf{U}}_k^t, \tilde{\mathbf{U}}_{c'}^{t+1}) \text{ with Dist} = \text{SAD or SGD}.$$

To use SGD, the gradient at each pixel $\vec{G}(i,j)$ is first computed using a 3×3 Sobel kernel, then the SGD of two patches within the matching area S is

$$\mathrm{SGD}(\tilde{\mathbf{U}}_k^t, \tilde{\mathbf{U}}_{c'}^{t+1}) = \sum_{(i,j) \in S} \arccos \left(\frac{\vec{G}_k^t(i,j) \cdot \vec{G}_{c'}^{t+1}(i,j)}{||\vec{G}_k^t(i,j)||\ ||\vec{G}_{c'}^{t+1}(i,j)||} \right).$$

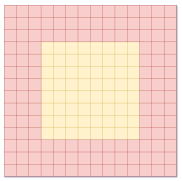

Fig. 1. Surrounding pixels (in red) as matching area S, and central block (in yellow) of size $w \times w$ for noise estimation.

The quantization technique of [22] and integral image technique are applied to accelerate the gradient-based matching.

However, simply processing block matching within the $w \times w$ blocks will lead to an underestimation of noise in the block difference. Indeed, if several candidate blocks contain the same signal as $\mathbf{U}_k^t$, then the block matching will try to find the block that also minimizes the noise difference. To avoid this, we separate the pixels used for block matching from those used for noise estimation. We use the surrounding pixels of the block as matching area S to find the best-matching block (see Fig. 1). Finally, a set of matched block pairs between two frames is obtained.

Block pair partitioning Since we are estimating the NLF of a signal-dependent noise, we group the blocks by their intensities; the noise levels are estimated accordingly. We partition the matched blocks into b bins of same size depending on the mean intensities of the block pairs. Given that noise is clipped in saturated pixels, using saturated pixels would lead to noise underestimation. Hence, we discard the block pairs with at least one saturated pixel before grouping them into bins.

Noise estimation For each bin of block pairs $B = \{(\tilde{\mathbf{U}}_k^t, \tilde{\mathbf{U}}_{(k)}^{t+1})\}$ where (k) is the index of the block in frame $t+1$ matched to $\tilde{\mathbf{U}}_k^t$, the mean intensity of all the blocks is denoted by $I_B \in \mathbb{R}$.

Next we estimate the noise level from the difference blocks $\{\mathbf{d}_k = \tilde{\mathbf{U}}_k^t - \tilde{\mathbf{U}}_{(k)}^{t+1}, k = 1, \dots, |B|\}$, with a prefilter step to select the difference blocks whose signals are well removed by subtraction. This is necessary because block matching cannot output perfectly-matched pairs due to noise, sampling, or changes in the scene content over time. The DCT-II is performed on the difference blocks to get the transformed blocks $\{\mathbf{D}_k = \mathrm{DCT}(\mathbf{d}_k) \in \mathbb{R}^{w \times w}\}$. Then each block $\mathbf{D}_k$ is divided into low and high frequency components by introducing a threshold T: $\mathbf{D}_k(i,j)$ is considered as low frequency if $i + j \leq T$ or else as high frequency. The low-frequency energy of $\mathbf{D}_k$ is computed as

$$\mathbf{V}_k^L = \sum_{i=1}^{w} \sum_{j=1}^{w} [\mathbf{D}_k(i,j)]^2 \, \mathbb{1}_{\{i+j \leq T\}},$$

where $\mathbb{1}_{\{\cdot\}}$ is the indicator function. A percentile q of difference blocks with the lowest low-frequency energies are selected as signal-free blocks, noted as $B' = \{\mathbf{D}_{[k]}, k = 1, \dots, \lfloor q|B| \rfloor\}$, where $[k]$ are the indices of the blocks sorted by increasing values of $\mathbf{V}_k^L$. The variance of high frequency

coefficients is calculated across the signal-free blocks by

$$\mathbf{V}^H(i,j) = \frac{1}{|B'|}\sum_{k=1}^{|B'|}[\mathbf{D}_k(i,j)]^2,$$

for $i+j>T$. Finally, the noise variance of the bin n is estimated by $V_n = \frac{1}{2}\text{median}(\{V^H(i,j), i+j>T\})$. Here the median value is halved since the noise of a block difference has double the original variance. The estimates of all the bins give us a sequence of intensities $\{I_n\}$ and the corresponding noise variances $\{V_n\}$.

The assumption behind the proposed method, inherited from [6], is that visual signals usually have more low frequency components than high frequency components. By selecting the lowest low-frequency-energy difference blocks we are selecting the blocks having the least residual signal. For these blocks, the influence of the signal on the high-frequency components will be the smallest.

4. EXPERIMENTS

To evaluate the performance of the proposed method, we conducted experiments on three datasets: the CRVD indoor dataset [23], the Colom dataset [24], and a dataset of synthetic drone videos (Synth-Drone). The datasets contain low-noise images or videos. For the CRVD indoor dataset, one image was taken from each of the 11 scenes. These images are mostly low textured. The Colom dataset contains 16 noiseless images captured from various scenes, some of which highly textured. The synthetic drone dataset consists of 15 videos captured by moving drones in bird's eye view (see Fig. 2).

We added intensity-dependent simulated noise with a

Fig. 2. Some test images of the CRVD indoor (1st row), Colom (2nd row) and Synth-Drone (3rd row) datasets. For color images, each channel is evaluated independently, resulting in one NLF per channel.

NLF of the form $g(I) = a + b \times I$ (see Sec. 3) as ground truth for evaluation. We used three noise models, $(a,b) \in \{(0.2,0.2),(0.8,0.8),(3.2,3.2)\}$, to simulate the noise level at different ISO settings, which cover noise standard deviations between 0.5 (the darkest signal) and 28 (the brightest signal). For each noise model, a sequence of 20 noisy frames was generated from a clean image or from a video sequence. Random jitter in the range of ± 2 pixel had been simulated on each noiseless frame to simulate more realistic videos.

The parameters of the proposed method selected as the best ones are: block size for noise estimation $w = 8$, thickness of surrounding area for block matching is 3, size of search window is 11×11, number of bins $b = 16$, threshold for low and high frequency $T = 5$, percentile of blocks with lowest low-frequency energy $q = 5\%$. We compared with seven other methods: [1], [5], [7], [9], [10], [12], [16]. Buades et al. [1]

Method	CRVD			Colom			Synth-Drone		
	$a=0.2$ $b=0.2$	$a=0.8$ $b=0.8$	$a=3.2$ $b=3.2$	$a=0.2$ $b=0.2$	$a=0.8$ $b=0.8$	$a=3.2$ $b=3.2$	$a=0.2$ $b=0.2$	$a=0.8$ $b=0.8$	$a=3.2$ $b=3.2$
Frame pair-wise estimates									
Proposed SAD (3.36s)	3.8	2.2	2.2	8.3	8.3	11.7	5.5	2.9	3.0
Proposed SGD (3.35s)	**2.5**	**1.7**	**2.0**	**6.2**	**6.4**	**9.4**	3.4	**1.8**	**1.6**
Ponomarenko [7] (0.28s)	3.1	2.2	2.1	18.0	9.7	13.1	**2.9**	2.2	2.1
Percentile [5] (1.40s)	8.7	8.1	7.2	28.5	14.0	12.0	6.5	4.5	4.0
Pyatykh [9] (0.28s)	8.0	7.2	15.0	36.8	19.8	21.2	23.5	36.6	51.7
Liu [10] (0.83s)	3.3	4.4	5.0	12.7	8.8	10.6	4.9	4.0	4.1
Amer [12] (0.07s)	9.1	7.7	6.7	14.6	11.0	10.8	9.7	7.0	6.9
Xiao [16] (3.61s)	5.3	8.2	11.3	8.0	9.9	14.8	5.6	7.4	10.1
Global estimates									
Proposed SAD (64.00s)	3.5	1.7	1.9	7.3	7.5	11.1	5.3	2.7	2.9
Proposed SGD (63.96s)	**2.3**	**1.2**	1.6	**5.8**	**6.0**	**9.1**	3.2	1.5	**1.4**
Buades [1] (8.93s)	3.7	1.9	**1.5**	17.9	8.9	12.1	**2.2**	**1.4**	1.6

Table 1. Mean relative errors on the three datasets (unit: %). "Proposed SAD" and "Proposed SGD" use respectively SAD and SGD metric for block matching. The best results are in bold red, and the second best in underlined blue. The computational time of each method per noise curve estimate is also indicated, benchmarked on a machine with 8 CPUs of 3.8GHz and 8GB memory for the Synth-Drone dataset.

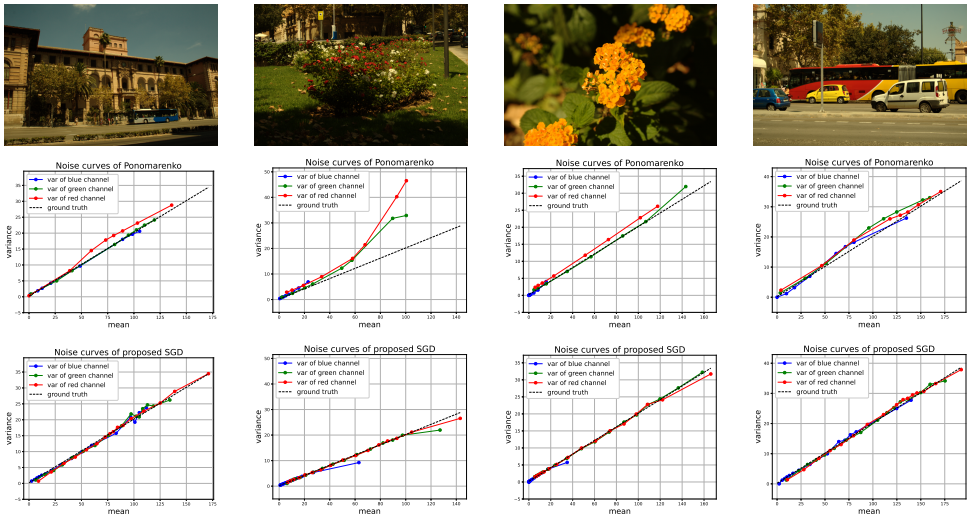

Fig. 3. Noise curves for red, green and blue channels estimated by Ponomarenko and the proposed method with SGD matching. The images are from Colom dataset, respectively from left to right: `building2`, `flowers1`, `flowers2` and `traffic`.

uses all the frames of a sequence to estimate the NLF, while the other methods, including ours, only use one frame or the difference of two frames for estimation. In order to do a fair comparison with Buades' method, we computed a global NLF from all the inter-frame estimates of our method. To do so, for each channel, given the 19 inter-frames NLF estimates, $\{(I_n^{(k)}, V_n^{(k)})\}$ where n denotes the n-th bin and k the k-th frame pair, we first computed the median intensity for each n-th bin over the 19 frame pairs, $I_n^{md} = \text{median}_k I_n^{(k)}$. Then, for each of the 19 NLFs, we computed the variances corresponding to $\{I_n^{md}\}$ by interpolation. Finally, $\{V_n^{md}\}$ was obtained as the median of these 19 interpolated values. For the rest of the methods, we compared the frame pair-wise estimates.

The mean relative error (MRE) was used to measure the accuracy of an estimated noise curve, $\text{MRE} = \frac{1}{b}\sum_{n=1}^{b} \frac{|V_n - g(I_n)|}{g(I_n)}$ with $\{I_n\}$ and $\{V_n\}$ the intensities and noise variances of a noise curve. The reason for using the relative error instead of the absolute error was to make a fair comparison of the noise estimates on the whole curve. Indeed, noise levels at low intensities usually have smaller values and thus smaller absolute errors whereas the relative errors can be high. The mean of MREs over different sequences, color channels and frame pairs was evaluated for frame pair-wise estimates. For global noise curve estimates, the mean of MREs over the sequences and color channels was evaluated. The results in Tab. 1 show that the proposed method using the SGD metric outperforms or performs as well as the existing SOTA methods in most cases, especially in the highly textured Colom dataset. Since the proposed method is a variant of Ponomarenko method, we focus on the comparison with it. For the CRVD dataset, where images are less textured, the use of difference blocks only provides a slight improvement over the use of raw blocks because there are enough homogeneous blocks in the image itself for Ponomarenko method to estimate noise in high frequencies. However, in the textured Colom dataset the high-frequency signals in the blocks greatly degrades the noise estimation.

Example noise curves of some images of the Colom dataset are shown in Fig. 3. We can see that Ponomarenko method usually overestimates the noise variance, particularly for `flowers1` where very few blocks are smooth. The strong textures in the selected blocks lead to a large bias of noise variance estimates. On the other hand, our method is able to suppress the textured signal by estimating within the difference of matched blocks, which leads to a more accurate noise curve. Note that the Ponomarenko method outputs "smoother" noise curves compared to ours. This is due to its filtering step, which attenuates the impact of outlying estimates while our method preserves the raw estimations.

5. CONCLUSION

In the proposed method, block matching is used to suppress signal in block differences, and is combined with Ponomarenko's principle of sorting blocks by their low-frequency energies and estimating noise in the DCT high frequencies of difference blocks. Using the SGD gradient direction matching metric outperforms the compared methods in accuracy.

Nevertheless, textured blocks present a wider range of intensities, hence, the pixel-wise Gaussian noise for these blocks is not homoscedastic. In this case, noise in high and low frequencies will be correlated. Then, selecting the blocks with the lowest energy in low-frequencies will cause a slight underestimation of the noise measured in the high-frequencies. We plan to further study this effect in the future.

In addition, a fast patch matching approaches such as [25] could be used for computational acceleration. Besides, our method can be complementary to the blind denoisers trained on homoscedastic noise [26] by applying variance stabilisation transform with our estimated NLF, so that the noise model satisfies the homoscedastic hypothesis.

References

[1] A. Buades and J. L. Lisani, "Denoising of noisy and compressed video sequences," in *VISIGRAPP*, 2017.

[2] M. Gardella, P. Musé, J.-M. Morel, and M. Colom, "Noisesniffer: A fully automatic image forgery detector based on noise analysis," in *IWBF*, IEEE, 2021, pp. 1–6.

[3] S.-C. Tai and S.-M. Yang, "A fast method for image noise estimation using Laplacian operator and adaptive edge detection," in *2008 3rd International Symposium on Communications, Control and Signal Processing*, IEEE, 2008, pp. 1077–1081.

[4] V. A. Pimpalkhute, R. Page, A. Kothari, K. M. Bhurchandi, and V. M. Kamble, "Digital image noise estimation using DWT coefficients," *IEEE Trans. on Image Processing*, vol. 30, pp. 1962–1972, 2021.

[5] M. Colom and A. Buades, "Analysis and extension of the Percentile method, estimating a noise curve from a single image," *Image Processing On Line*, vol. 3, pp. 332–359, 2013.

[6] N. Ponomarenko, V. Lukin, M. Zriakhov, A. Kaarna, and J. Astola, "An automatic approach to lossy compression of AVIRIS images," in *IGARSS*, IEEE, 2007, pp. 472–475.

[7] M. Colom and A. Buades, "Analysis and extension of the Ponomarenko et al. method, estimating a noise curve from a single image," *Image Processing On Line*, vol. 3, pp. 173–197, 2013.

[8] S. Mohan, T. Raghavendiran, and R. Rajavel, "Patch based fast noise level estimation using DCT and standard deviation," *Cluster Computing*, vol. 22, no. 6, pp. 14 495–14 504, 2019.

[9] S. Pyatykh, J. Hesser, and L. Zheng, "Image noise level estimation by principal component analysis," *IEEE trans. on image processing*, vol. 22, no. 2, pp. 687–699, 2012.

[10] X. Liu, M. Tanaka, and M. Okutomi, "Single-image noise level estimation for blind denoising," *IEEE trans. on image processing*, vol. 22, no. 12, pp. 5226–5237, 2013.

[11] M. Colom, A. Buades, and J.-M. Morel, "Nonparametric noise estimation method for raw images," *Journal of the Optical Society of America A*, vol. 31, no. 4, pp. 863–871, 2014.

[12] M. Rakhshanfar and M. A. Amer, "Estimation of gaussian, poissonian–gaussian, and processed visual noise and its level function," *IEEE Trans. on Image Processing*, vol. 25, no. 9, pp. 4172–4185, 2016.

[13] Z. Lei and X. Peixia, "Differential video noise estimation," in *2006 International Conference on Communication Technology*, 2006, pp. 1–4.

[14] A. Amer and E. Dubois, "Fast and reliable structure-oriented video noise estimation," *Circuits and Systems for Video Technology, IEEE Trans. on*, vol. 15, pp. 113–118, Feb. 2005.

[15] W. Yin and H. Zhao, "A novel method of video noise estimation based on motion estimation," in *2012 IEEE ICSPCC*, 2012, pp. 295–299.

[16] J. Xiao, H. Tian, Y. Zhang, Y. Zhou, and J. Lei, "Blind video denoising via texture-aware noise estimation," *Computer Vision and Image Understanding*, vol. 169, pp. 1–13, 2018.

[17] M. Ghazal, A. Amer, and A. Ghrayeb, "Structure-oriented spatio-temporal video noise estimation," in *IEEE ICASSP Proceedings*, vol. 2, 2006, pp. II–II.

[18] M. Ghazal, A. Amer, and A. Ghrayeb, "A real-time technique for spatio–temporal video noise estimation," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 17, no. 12, pp. 1690–1699, 2007.

[19] V. Zlokolica, A. Pizurica, and W. Philips, "Noise estimation for video processing based on spatio-temporal gradients," *IEEE Signal Processing Letters*, vol. 13, no. 6, pp. 337–340, 2006.

[20] S.-M. Yang and S.-C. Tai, "A fast and reliable algorithm for video noise estimation based on spatio-temporal sobel gradients," in *InECCE 2011*, 2011, pp. 191–195.

[21] M. Izadi, N. Birkbeck, and B. Adsumilli, "Mutual noise estimation algorithm for video denoising," in *2019 IEEE ICIP*, 2019, pp. 2424–2428.

[22] S. Hinterstoisser, C. Cagniart, S. Ilic, *et al.*, "Gradient response maps for real-time detection of textureless objects," *IEEE trans. on pattern analysis and machine intelligence*, vol. 34, no. 5, pp. 876–888, 2011.

[23] H. Yue, C. Cao, L. Liao, R. Chu, and J. Yang, "Supervised raw video denoising with a benchmark dataset on dynamic scenes," in *Proceedings of the IEEE/CVF CVPR*, 2020, pp. 2301–2310.

[24] M. Colom, *Noise-free test images dataset*. [Online]. Available: http://mcolom.info/pages/no_noise_images/.

[25] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, 2009.

[26] S. Mohan, Z. Kadkhodaie, E. P. Simoncelli, and C. Fernandez-Granda, "Robust and interpretable blind image denoising via bias-free convolutional neural networks," *arXiv preprint arXiv:1906.05478*, 2019.