FastDVDnet: Towards Real-Time Video Denoising Without Explicit Motion Estimation

Matias Tassano, Julie Delon, Thomas Veit

FastDVDnet: Towards Real-Time Video Denoising Without Explicit Motion Estimation

Matias Tassano, Julie Delon, and Thomas Veit

***Abstract*—In this paper, we propose a state-of-the-art video denoising algorithm based on a convolutional neural network architecture. Until recently, video denoising with neural networks had been a largely under explored domain, and existing methods could not compete with the performance of the best patch-based methods. The approach we introduce in this paper, called FastDVDnet, shows similar or better performance than other state-of-the-art competitors with significantly lower computing times. In contrast to other existing neural network denoisers, our algorithm exhibits several desirable properties such as fast run-times, and the ability to handle a wide range of noise levels with a single network model. The characteristics of its architecture make it possible to avoid using a costly motion compensation stage while achieving excellent performance. The combination between its denoising performance and lower computational load makes this algorithm attractive for practical denoising applications. We compare our method with different state-of-art algorithms, both visually and with respect to objective quality metrics.**

***Index Terms*—video denoising, CNN, residual learning, neural networks, image restoration, end-to-end training**

I. Introduction

DESPITE the immense progress made in recent years in photographic sensors, noise reduction remains an essential step in video processing, especially when shooting conditions are difficult (low light, small sensors, etc.).

Although image denoising has remained a very active research field through the years, too little work has been devoted to the restoration of digital videos. It should be noted however that some crucial aspects differentiate these two problems. On one hand, a video contains much more information than a still image, which could help in the restoration process. On the other hand, video restoration requires good temporal coherency, which makes the restoration process much more demanding. On top of this, since all recent cameras produce videos in high definition or larger, very fast and efficient algorithms are needed.

In [1] we introduced a state-of-the-art network for Deep Video Denoising: DVDnet. The performance of this algorithm compares favorably to other methods; its outputs present remarkable temporal coherence, low flickering, and accurate detail preservation. Although it runs significantly faster than patch-based state-of-the-art algorithms, its running times are constrained by the time it takes to estimate motion in temporal neighboring frames. Running times could be greatly reduced if we disposed of the motion estimation stage altogether. However, motion estimation and the quality of its results play a crucial role in a successful video denoising algorithm.

In this paper we introduce another network for deep video denoising: FastDVDnet. This algorithm features a number of important changes with respect to its predecessor. Most notably, instead of employing an explicit motion estimation stage, the algorithm is able to implicitly handle motion thanks to the traits of its architecture. This results in a state-of-the-art algorithm which outputs high quality denoised videos while featuring very fast running times—even thousands of times faster than other relevant methods.

A. Image Denoising

Contrary to video denoising, image denoising has enjoyed consistent popularity in past years. A myriad of new image denoising methods based on deep learning techniques have drawn considerable attention due to their outstanding performance. Schmidt and Roth proposed in [2] the cascade of shrinkage fields method that unifies the random field-based model and half-quadratic optimization into a single learning framework. Based on this method, Chen and Pock proposed in [3] a trainable nonlinear reaction diffusion model. This model can be expressed as a feed-forward deep network by concatenating a fixed number of gradient descent inference steps. Methods such as these two achieve performances comparable to those of well-known patch-based algorithms such as BM3D [4] or non-local Bayes (NLB [5]). However, their performance is restricted to specific forms of prior. Additionally, many hand-tuned parameters are involved in the training process. In [6], a multi-layer perceptron was successfully applied for image denoising. Nevertheless, a significant drawback of all these algorithms is that a specific model must be trained for each noise level.

Another widespread approach involves the use of convolutional neural networks (CNN), e.g. RBDN [7], MWCNN [8], DnCNN [9], and FFDNet [10]. Their performance compares favorably to other state-of-the-art image denoising algorithms, both quantitatively and visually. These methods are composed of a succession of convolutional layers with nonlinear activation functions in between them. This type of architecture has been applied to the problem of joint denoising and demosaicing of RGB and raw images by Gharbi et al. in [11], whereas [12], [13] approach a similar problem but for low-light conditions. In [8], Liu et al. fuse a multi-level wavelet transform with a modified U-Net [14] network, and apply this architecture to different image reconstruction applications. A salient feature that these CNN-based methods present is the ability to denoise several levels of noise with only one trained

M. Tassano is with the MAP5 Laboratory, Paris Descartes University, 75006 Paris, France, and with GoPro Technology France e-mail: matias.tassano@parisdescartes.fr.

J. Delon is with the MAP5 Laboratory, Paris Descartes University, 75006 Paris, France.

T. Veit is with GoPro Technology France.

model. Proposed by Zhang et al. in [9], DnCNN is an end-to-end trainable deep CNN for image denoising. This method is able to denoise different noise levels (e.g. with standard deviation $\sigma \in [0, 55]$) with only one trained model. One of its main features is that it implements residual learning [15], i.e. it estimates the noise existent in the input image rather than the denoised image. In a following paper [10], Zhang et al. proposed FFDNet, which builds upon the work done for DnCNN. The main difference of FFDNet with respect to DnCNN is the fact that almost all the denoising processing is performed at a quarter-resolution, without compromising the quality of its results [10], [16].

B. Video Denoising

Video denoising is much less explored in the literature. The majority of recent video denoising methods are patch-based. For instance, Kokaram et al. proposed in [17] a 3D Wiener filtering scheme. We note in particular an extension of the popular BM3D to video denoising, V-BM4D [18], and Video non-local Bayes (VNLB [19]). Neural network methods for video denoising have been even rarer than patch-based approaches. The algorithm in [20] by Chen et al. is one of the first to approach this problem with recurrent neural networks. However, their algorithm only works on grayscale images and it does not achieve satisfactory results, probably due to the difficulties associated with training recurring neural networks [21]. Vogels et al. proposed in [22] an architecture based on kernel-predicting neural networks able to denoise Monte Carlo rendered sequences. The Video Non-Local Network (VNLnet [23]) fuses a CNN with a self-similarity search strategy. For each patch, the network finds the most similar patches via its first non-trainable layer, and this information is later used by the CNN to predict the clean image. In [1] Tassano et al. proposed DVDnet, which splits the denoising of a given frame in two separate denoising stages. Like several other methods, it relies on the estimation of motion of neighboring frames. Nowadays, the state-of-the-art is defined by DVDnet, VNLnet and VNLB. VNLB and VNLnet show the best performances for small values of noise, while DVDnet yields better results for larger values of noise. Both DVDnet and VNLnet feature significantly faster inference times than VNLB. As we will see, the performance of the method we introduce in this paper compares to the performance of the state-of-the-art, while featuring even faster runtimes.

II. FastDVDnet

For video denoising algorithms, temporal coherence and the lack of flickering are crucial aspects in the perceived quality of the results [24], [25]. In order to achieve these, an algorithm must make use of the temporal information existent in neighboring frames when denoising a given frame of an image sequence. In general, most previous approaches based on deep learning have failed to employ this temporal information effectively. Successful state-of-the-art algorithms rely mainly on two factors to enforce temporal coherence in the results, namely the extension of search regions from spatial neighborhoods to volumetric neighborhoods, and the use of motion estimation.

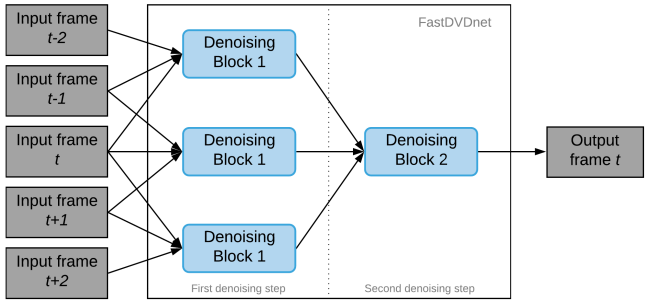
The use of volumetric, or spatio-temporal, neighborhoods implies that when denoising a given pixel (or patch), the algorithm is going to look for similar pixels (patches) not only in the reference frame, but also in adjacent frames of the sequence. The benefits of this are two-fold. First, the temporal neighbors provide additional information which can be used to denoise the reference frame. Second, using temporal neighbors helps to reduce flickering as the residual error in each frame will be correlated.

Videos feature a strong temporal redundancy along motion trajectories. This fact should facilitate denoising videos with respect to denoising images. Yet, this added information in the temporal dimension also creates an extra degree of complexity which could be difficult to tackle. In this context, motion estimation and/or compensation has been employed in a number of video denoising algorithms to help to improve denoising performance and temporal consistency [1], [18], [19], [26], [27].

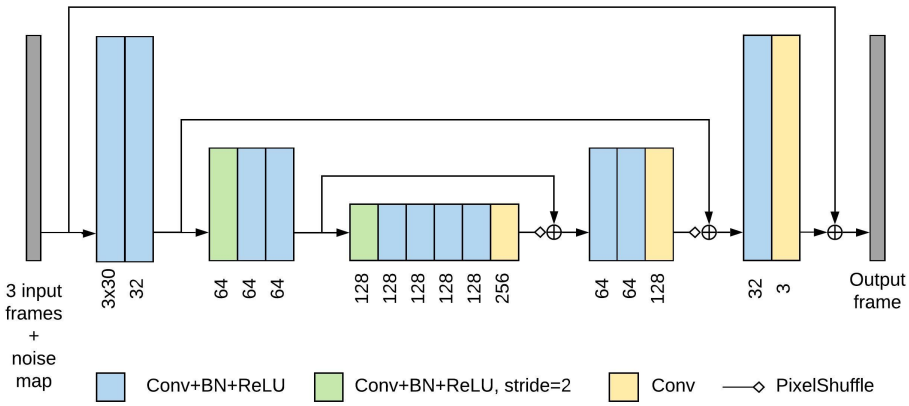
We thus incorporated these two elements into our architecture. However, our algorithm does not include an explicit motion estimation/compensation stage. The capacity of handling the motion of objects is inherently embedded into the proposed architecture. Indeed, our architecture is composed of a number of modified U-Net [14] blocks (see section II-A for more details about these blocks). Multi-scale, U-Net-like architectures have been shown to have the ability to learn misalignment [28], [29]. Our cascaded architecture increases this capacity of handling movement even further. In contrast to [1], our architecture is trained end-to-end without optical flow alignment, which avoids distortions and artifacts due to erroneous flow. As a result, we are able to eliminate a costly dedicated motion compensation stage without sacrificing performance. This leads to an important reduction of runtimes: our algorithm runs three orders of magnitude faster than VNLB, and an order of magnitude faster than DVDnet and VNLnet.

Figure 1a displays a diagram of the architecture of our method. When denoising a given frame at time t, $\tilde{\mathbf{I}}_t$, its $2T = 4$ neighboring frames are also taken as inputs. That is, the inputs of the algorithm will be $\left\{\tilde{\mathbf{I}}_{t-2}, \tilde{\mathbf{I}}_{t-1}, \tilde{\mathbf{I}}_t, \tilde{\mathbf{I}}_{t+1}, \tilde{\mathbf{I}}_{t+2}\right\}$. The model is composed of different spatio-temporal denoising blocks, assembled in a cascaded two-step architecture. These denoising blocks are all similar, and consist of a modified U-Net model which takes three frames as inputs. The three blocks in the first denoising step share the same weights, which leads to a reduction of memory requirements of the model and facilitates the training of the network. Similar to [10], [11], a noise map is also included as input, which allows the processing of spatially varying noise [16]. Contrary to other denoising algorithms, our denoiser takes no other parameters as inputs apart from the image sequence and the estimation of the input noise.

Observe that experiments presented in this paper focus on the case of additive white Gaussian noise (AWGN). Nevertheless, this algorithm can be straightforwardly extended to other



(a)



(b)

Fig. 1. *Architecture used in FastDVDnet*. (a) A high-level diagram of the architecture. Five consecutive frames are used to denoise the middle frame. The frames are taken as triplets of consecutive frames and input to the *Denoising Blocks 1*. The instances of these blocks have all the same weights. The triplet composed by the outputs of these blocks are used as inputs for *Denoising Block 2*. The output of the latter is the estimate of the central input frame (*Input frame t*). Both *Denoising Block 1* and *Denoising Block 2* share the same architecture, which is shown in (b). The denoising blocks of FastDVDnet are composed of a modified multi-scale U-Net.

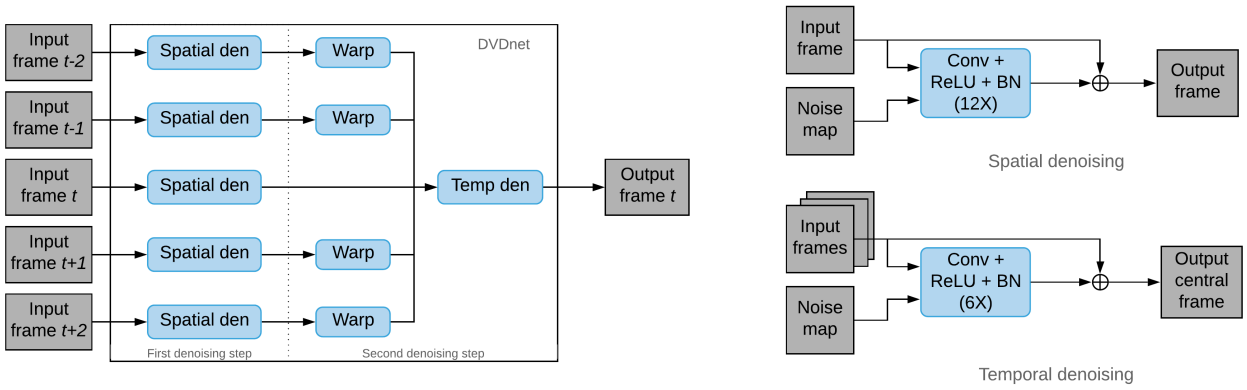


Fig. 2. *Architecture of DVDnet* Each of the five input frames is individually denoised with a spatial single-scale denoiser, in the first denoising step. The temporal neighbors are motion-compensated with respect to the central frame, and these 5 frames are concatenated and input to the temporal denoising block.

types of noise, e.g. spatially varying noise (e.g. Poissonian). Let $\mathbf{I}$ be a noiseless image, while $\tilde{\mathbf{I}}$ is its noisy version corrupted by a realization of zero-mean white Gaussian noise $\mathbf{N}$ of standard deviation σ, then

$$\tilde{\mathbf{I}} = \mathbf{I} + \mathbf{N} \ . \quad (1)$$

A. Denoising Blocks

Both denoising blocks displayed in fig. 1a, *Denoising Block 1* and *Denoising Block 2*, consist of a modified U-Net architecture. All the instances of *Denoising Block 1* share the same weights. U-Nets are essentially a multi-scale encoder-

decoder architecture, with skip-connections [15] that forward the output of each one of the encoder layers directly to the input of the corresponding decoder layers. A more detailed diagram of these blocks is shown in fig. 1b. Our denoising blocks present some differences with respect to the standard U-Net:

- The encoder has been adapted to take three frames and a noise map as inputs
- The upsampling in the decoder is performed with a *PixelShuffle* layer [30], which helps reducing gridding artifacts
- The merging of the features of the encoder with those of the decoder is done with a pixel-wise addition operation instead of a channel-wise concatenation. This results in a reduction of memory requirements
- Blocks implement residual learning—with a residual connection between the central noisy input frame and the output—, which has been observed to ease the training process [16]

The design characteristics of the denoising blocks make a good compromise between performance and fast running times. These denoising blocks are composed of a total of $D = 16$ convolutional layers. In most layers, the outputs of its convolutional layers are followed by point-wise *ReLU* [31] activation functions $ReLU(\cdot) = \max(\cdot, 0)$, except for the last layer. At training time, batch normalization layers (*BN* [32]) are placed between the convolutional and *ReLU* layers. At evaluation time, the batch normalization layers are removed, and replaced by an affine layer that applies the learned normalization.

B. Comparison to DVDnet [1]

Similarly to FastDVDnet, DVDnet features a two-step cascaded architecture, shown in fig. 2 for the sake of comparison. This allows DVDnet to also make use of the information existent in temporal neighboring frames. Nevertheless, input frames are processed individually in the first denoising step by the spatial denoising blocks, that is, there is no sharing of temporal information in this step. After the first step, the pre-denoised temporal neighbors are aligned with respect to the central frame at the motion compensation stage. Then, the five frames are concatenated and input to the temporal denoising block. Generally speaking, the spatial and temporal blocks consist of modified, single-scale FFDNet networks. They are composed of $D_{spa} = 12$, and $D_{temp} = 6$ convolutional layers, respectively. The number of feature maps is set to $W = 96$. In both blocks, inputs are first downscaled to a quarter-resolution, and the upscaling back to full resolution is performed with a *PixelShuffle* layer.

One of the main differences in FastDVDnet with respect to DVDnet resides in the absence of a motion estimation stage. DVDnet employs the DeepFlow algorithm [33] for the estimation of the optical flow between frames. Although there exist faster motion estimation schemes, such as the Farnebäck algorithm [34] or LiteFlowNet [35], we observed in our tests that the quality of the flow estimation greatly impacts on the quality of the denoising results, and that DVDnet performed best with DeepFlow than with other of the tested algorithms. Considering this and the fact that flow estimation stands for the large majority of the running times, we decided to discard motion estimation in FastDVDnet altogether. However, we needed to introduce a number of techniques to handle motion and to effectively employ temporal information instead. These techniques are discussed further in section III.

III. Discussion

A. Two-step denoising

As mentioned in section II, both DVDnet and FastDVDnet feature a cascaded, two-step denoising architecture. The motivation behind this is to effectively employ the information existent in the temporal neighbors, and to enforce the temporal correlation of the remaining noise in output frames. To prove that the two-step denoising is a necessary feature, we conducted the following experiment: we modified a *Denoising Block* of FastDVDnet (see fig. 1b) to take five frames as inputs instead of three, which we will refer to as *Den_Block_5inputs*. In this way, the same amount of temporal neighboring frames are considered and the same information as in FastDVDnet is processed by this new denoiser. A diagram of the architecture of this model is shown in fig. 3. We then trained this new model and compared the results of denoising of sequences against the results of FastDVDnet (see section IV for more details about the training process). Table I displays the PSNRs on four 854×480 color sequences for both denoisers. Note that for this test in particular, the *Denoising Blocks* of these two architectures do not implement residual learning. It can be seen that the cascaded architecture of FastDVDnet presents a clear advantage on *Den_Block_5inputs*. On top of this, results by *Den_Block_5inputs* present a sharp increase on temporal artifacts—flickering. Despite it being a multi-scale architecture, *Den_Block_5inputs* cannot handle the motion of objects in the sequences as well as the two-step architecture of FastDVDnet can.

In [36], Simonyan and Zisserman demonstrated that given a certain value of receptive field, a deeper network with small convolutional kernels will outperform shallower networks with larger kernels in classification tasks. Indeed, more layers imply a larger number of nonlinearities, which in turn leads to more powerful nonlinear mappings. Also, smaller kernels lead to less parameters, which can be seen as sort of regularization. Since then, using spatial kernels of small sizes has been the prevailing trend in neural networks, including architectures for image restoration. Our intuition is that here as well smaller cascaded 'temporal kernels' outperform larger kernels when dealing with temporal information.

B. Multi-scale architecture and end-to-end training

In order to investigate the importance of using multi-scale denoising blocks in our architecture, we conducted the following experiment: we modified the FastDVDnet architecture by replacing its *Denoising Blocks* by the denoising blocks of DVDnet, as displayed in fig. 2. This results in a two-step cascaded architecture, with single-scale denoising blocks, trained end-to-end, and with no compensation of motion in the scene. We will call this new architecture FastDVDnet_Single.

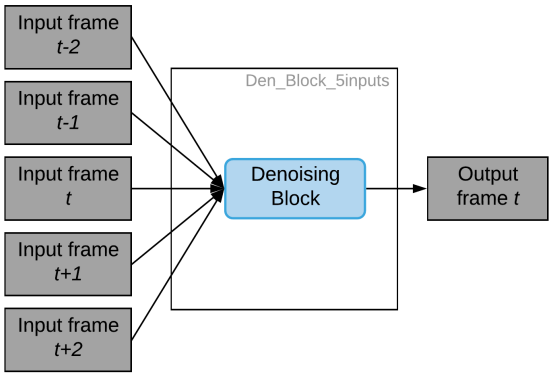


Fig. 3. Architecture of the *Den_Block_5inputs* denoiser.

TABLE I
COMPARISON OF $PSNR$ OF TWO DENOISERS ON A SET OF FOUR SEQUENCES. BEST RESULTS ARE SHOWN IN BOLD. NOTE: FOR THIS TEST IN PARTICULAR, NEITHER OF THESE DENOISERS IMPLEMENT RESIDUAL LEARNING.

		FastDVDnet	Den_Block_5inputs
$\sigma = 10$	hypersmooth	**37.34**	35.64
	motorbike	**34.86**	34.00
	rafting	**36.20**	34.61
	snowboard	**36.50**	34.27
$\sigma = 30$	hypersmooth	**32.17**	31.21
	motorbike	**29.16**	28.77
	rafting	**30.73**	30.03
	snowboard	**30.59**	29.67
$\sigma = 50$	hypersmooth	**29.77**	28.92
	motorbike	**26.51**	26.19
	rafting	**28.45**	27.88
	snowboard	**28.08**	27.37

Table II shows the PSNRs on four 854×480 color sequences for both FastDVDnet and FastDVDnet_Single. Note that for this test in particular, neither of these denoisers implement residual learning. It can be seen that the usage of multi-scale denoising blocks improves denoising results considerably.

We also experimented with training the denoising blocks in each step of FastDVDnet separately—as done in DVDnet. Although the results in this case certainly improved with respect to those of FastDVDnet_Single, a noticeable flickering remained in its outputs. Switching from this separate training to an end-to-end trained helped reduce temporal artifacts considerably.

C. Handling of Motion

Apart from the reduction of runtimes, avoiding the use of motion compensation by means of optical flow has an additional benefit. Video denoising algorithms that depend explicitly on motion estimation techniques often present artifacts due to erroneous flow in challenging cases, such as occlusions or strong noise. The different techniques discussed in this section—namely a multi-scale of the denoising blocks, the cascaded two-step denoising architecture, and end-to-end training—not only provide FastDVDnet the ability to handle motion, but also help avoid artifacts related to erroneous flow estimation. Also, and similarly to [1], [9], [16], the denoising blocks of FastDVDnet implement residual learning, which helps improving the quality of results a step further. Figure 4 shows an example on artifacts due to erroneous flow on three consecutive frames and of how the multi-scale architecture of FastDVDnet is able to avoid them.

TABLE II
COMPARISON OF $PSNR$ OF A SINGLE-SCALE DENOISER AGAINST A MULTI-SCALE DENOISER ON A SET OF FOUR SEQUENCES. BEST RESULTS ARE SHOWN IN BOLD. NOTE: FOR THIS TEST IN PARTICULAR, NEITHER OF THESE DENOISERS IMPLEMENT RESIDUAL LEARNING.

		FastDVDnet	FastDVDnet_Single
$\sigma = 10$	hypersmooth	**37.34**	36.61
	motorbike	**34.86**	34.30
	rafting	**36.20**	35.54
	snowboard	**36.50**	35.50
$\sigma = 30$	hypersmooth	**32.17**	31.54
	motorbike	**29.16**	28.82
	rafting	**30.73**	30.36
	snowboard	**30.59**	30.04
$\sigma = 50$	hypersmooth	**29.77**	29.14
	motorbike	**26.51**	26.22
	rafting	**28.45**	28.11
	snowboard	**28.08**	27.56

IV. TRAINING DETAILS

The training dataset consists of input-output pairs

$$P_t^j = \left\{ \left((\, S_t^j,\, \mathbf{M}^j \,),\, \mathbf{I}_t^j \right) \right\}_{j=0}^{m_t},$$

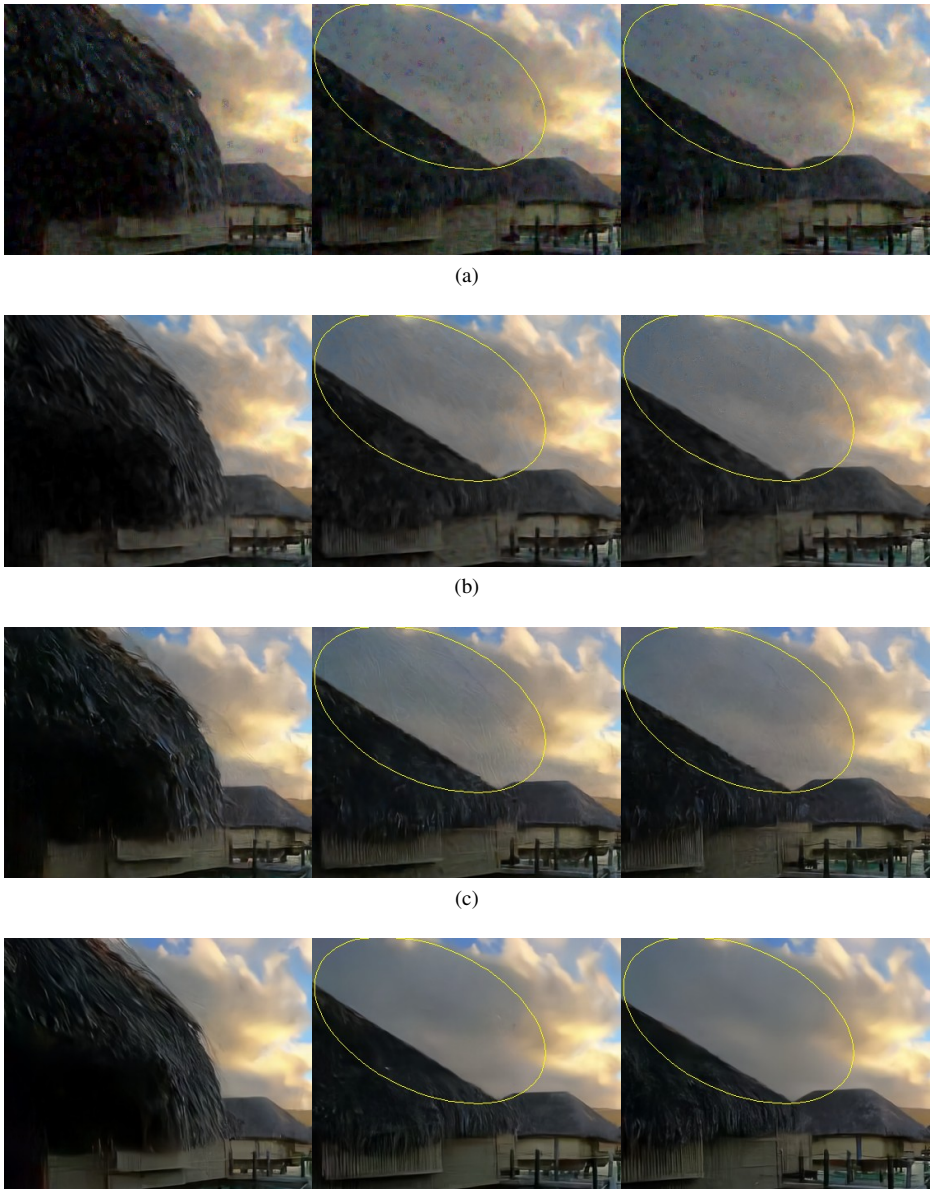
where $S_t^j = (\tilde{\mathbf{I}}_{t-2}^j,\, \tilde{\mathbf{I}}_{t-1}^j,\, \tilde{\mathbf{I}}_t^j,\, \tilde{\mathbf{I}}_{t+1}^j,\, \tilde{\mathbf{I}}_{t+2}^j)$ is a collection of $2T + 1 = 5$ spatial patches cropped at the same location in contiguous frames, and $\mathbf{I}^j$ is the clean central patch of the sequence. These are generated by adding AWGN of $\sigma \in [5,\, 50]$ to clean patches of a given sequence, and the corresponding noise map $\mathbf{M}^j$ is built in this case constant with all its elements equal to σ. Spatio-temporal patches are randomly cropped from randomly sampled sequences of the training dataset.

A total of $m_t = 384000$ training samples are extracted from the training set of the DAVIS database [37]. The spatial size of the patches is 96×96, while the temporal size is $2T + 1 = 5$. The loss function is

$$\mathcal{L}(\theta) = \frac{1}{2m_t} \sum_{j=1}^{m_t} \left\| \hat{\mathbf{I}}_t^j - \mathbf{I}_t^j \right\|^2, \tag{2}$$

where $\hat{\mathbf{I}}_t^j = \mathcal{F}((\, S_t^j,\, \mathbf{M}^j \,);\, \theta)$ is the output of the network, and θ is the set of all learnable parameters.

The architecture has been implemented in PyTorch [38], a popular machine learning library. The ADAM algorithm [39] is applied to minimize the loss function, with all its hyperparameters set to their default values. The number of epochs is set to 80, and the mini-batch size is 96. The scheduling of the learning rate is also common to both cases. It starts at $1\mathrm{e}{-3}$ for the first 50 epochs, then changes to $1\mathrm{e}{-4}$ for the following 10 epochs, and finally switches to $1\mathrm{e}{-6}$ for



(a) (b) (c) (d)

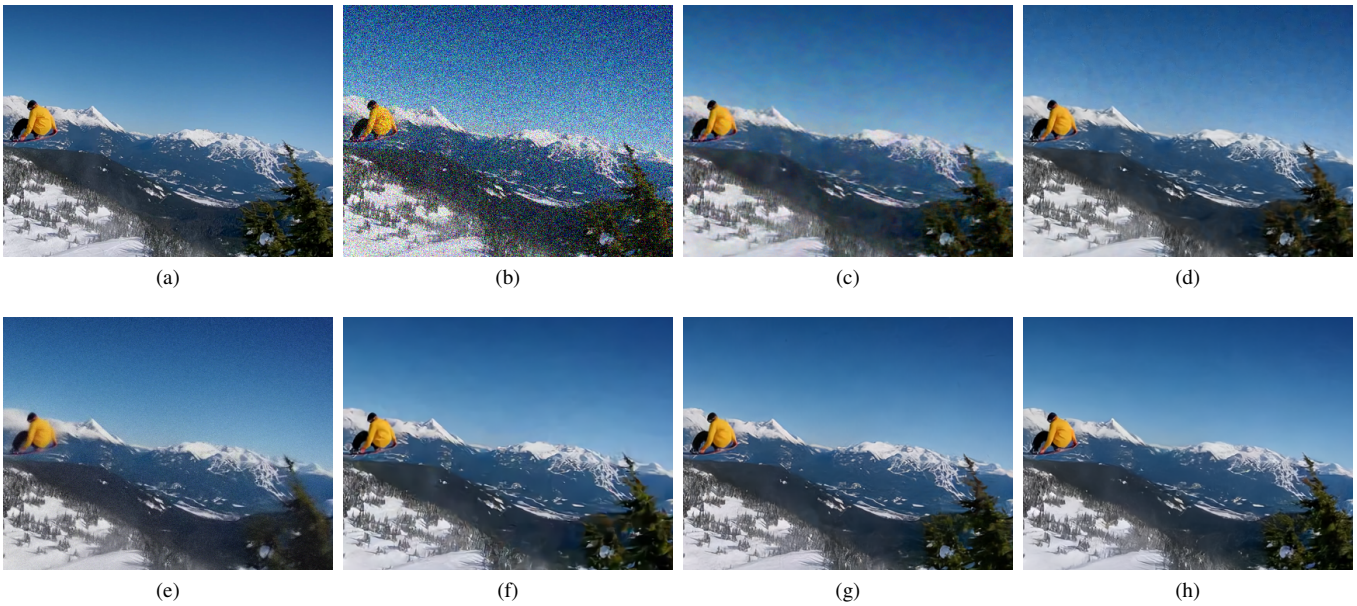
Fig. 4. *Motion artifacts due to occlusion*. Three consecutive frames of the results of the 'hypersmooth' sequence, $\sigma = 50$ (a) V-BM4D. (b) VNLB. (c) DVDnet. (d) FastDVDnet. Video denoising algorithms that depend explicitly on motion estimation techniques often present artifacts due to erroneous flow in challenging cases. In the example above, the occlusion of the front building leads to motion artifacts in the results of V-BM4D, VNLB, and DVDnet. Explicit motion compensation is avoided in the architecture of FastDVDnet. Indeed, the network is able to implicitly handle motion due to its design characteristics. Best viewed in digital format.

the remaining of the training. In other words, a learning rate step decay is used in conjunction with ADAM. The mix of learning rate decay and adaptive rate methods has also been applied to other deep learning projects [40], [41], usually with positive results. Data is augmented by introducing rescaling by different scale factors and random flips. During the first 60 epochs, the orthogonalization of the convolutional kernels is applied as a means of regularization. It has been observed that initializing the training with orthogonalization may be beneficial to performance [10], [16].

V. Results

Two different testsets were used for benchmarking our method: the DAVIS-test testset, and Set8, which is composed of 4 color sequences from the *Derf's Test Media collection*[1] and 4 color sequences captured with a GoPro camera. The DAVIS set contains 30 color sequences of resolution 854×480. The sequences of Set8 have been downscaled to a resolution of 960×540. In all cases, sequences were limited to a maximum of 85 frames. We used the DeepFlow algorithm to compute flow maps for DVDnet and VNLB. VNLnet requires models trained for specific noise levels. As no model is provided for $\sigma = 30$, no results are shown for this noise level in either of the tables. We also compare our method to a commercial blind denoising software, Neat Video (NV [42]). For NV, its automatic noise profiling settings were used to manually

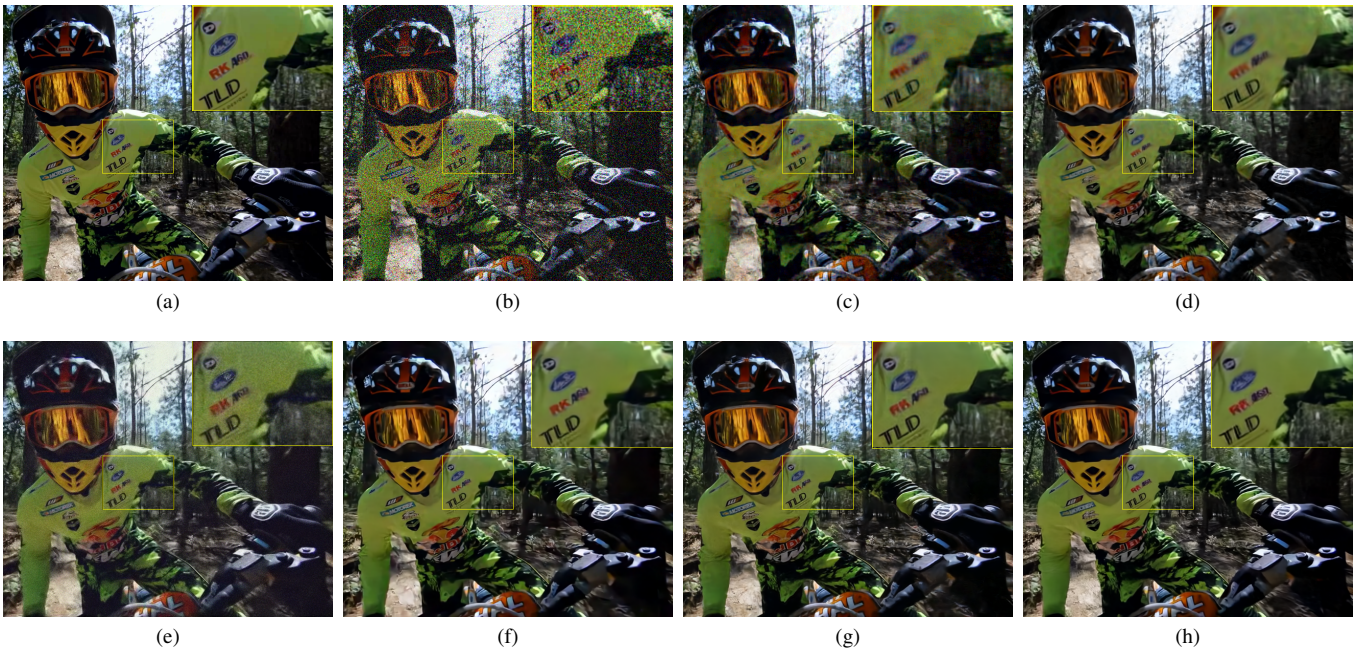
[1] https://media.xiph.org/video/derf



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 5. *Comparison of results of the 'snowboarding' sequence*. (a) Clean frame. (b) Noisy frame $\sigma = 40$. (c) V-BM4D. (d) VNLB. (e) NV. (f) VNLnet. (g) DVDnet. (h) FastDVDnet. Patch-based methods (V-BM4D, VNLB, and even VNLnet) struggle with noise in flat areas, such as the sky, and leave behind medium-to-low-frequency noise. This leads to results with noticeable flickering, as the remaining noise is temporally decorrelated. On the other hand, DVDnet and FastDVDnet output very convincing and visually pleasant results. Best viewed in digital format.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 6. *Comparison of results of the 'motorbike' sequence*. (a) Clean frame. (b) Noisy frame $\sigma = 50$. (c) V-BM4D. (d) VNLB. (e) NV. (f) VNLnet. (g) DVDnet. (h) FastDVDnet. Note the clarity of the denoised text, and the lack of low-frequency residual noise and chroma noise for FastDVDnet and DVDnet. Best viewed in digital format.

denoise the sequences of Set8. Note that values shown are the average for all sequences in the testset, the PNSR of a sequence is computed as the average of the PSNRs of each frame.

In general, both DVDnet and FastDVDnet output sequences which feature remarkable temporal coherence. Flickering rendered by our methods is notably small, especially in flat areas, where patch-based algorithms often leave behind low-frequency residual noise. An example can be observed in fig. 5 (which is best viewed in digital format). Temporally decorrelated low-frequency noise in flat areas appears as particularly annoying in the eyes of the viewer. More video examples can be found in the website of the algorithm. The reader is encouraged to watch these examples to compare the

visual quality of the results of our methods[2].

Patch-based methods are prone to surpass DVDnet and FastDVDnet in sequences with a large portion of repetitive structures as these methods exploit the non-local similarity prior. On the other hand, our algorithms handle non-repetitive textures very well, see e.g. the clarity of the denoised text and vegetation in fig. 6.

Tables III and IV show a comparison of PSNR and ST-RRED on the Set8 and DAVIS dataset, respectively. The Spatio-Temporal Reduced Reference Entropic Differences (ST-RRED) is a high performing reduced-reference video quality assessment metric [43]. This metric not only takes into account image quality, but also temporal distortions in the video. We computed the ST-RRED scores with the implementation provided by the *scikit-video* library[3].

It can be observed that for smaller values of noise, VNLB performs better on Set8. Indeed, DVDnet tends to over denoise in some of these cases. FastDVDnet and VNLnet are the best performing algorithms on DAVIS for small sigmas in terms of PSNR and ST-RRED, respectively. However, for larger values of noise DVDnet surpasses VNLB. FastDVDnet performs consistently well in all cases, which is a remarkable feat considering that it runs 80 times faster than DVDnet, 26 times faster than VNLnet, and more than 4000 times faster than VNLB (see section VI). Contrary to other denoisers based on CNNs—e.g. VNLnet—, our algorithms are able to denoise different noise levels with only one trained model. On top of this, the use of methods involve no hand-tuned parameters, since they only take the image sequence and the estimation of the input noise as inputs.

VI. Running times

Our method achieves fast inference times, thanks to its design characteristics and simple architecture. Our algorithm takes only $100ms$ to denoise a 960×540 color frame, which is more than 3 orders of magnitude faster than V-BM4D and VNLB, and more than an order of magnitude faster than other CNN algorithms which run on GPU, DVDnet and VNLnet. The algorithms were tested on a multi-core server with a Titan Xp NVIDIA GPU card. Table V compares the running times of different state-of-the-art algorithms.

VII. Conclusion

In this paper, we presented FastDVDnet, a state-of-the-art video denoising algorithm. Denoising results of FastDVDnet feature remarkable temporal coherence, very low flickering, and excellent detail preservation. The algorithm runs 80 times faster than its predecessor, DVDnet, and one to three orders of magnitude faster than other state-of-the-art competitors. In this sense, our approach proposes a major step forward towards high quality real-time video noise reduction. Although the results presented in this paper hold for Gaussian noise, our method could be extended to denoise other types of noise.

[2]https://github.com/m-tassano/fastdvdnet and https://github.com/m-tassano/dvdnet

[3]http://www.scikit-video.org

Acknowledgment

Julie Delon would like to thank the support of NVIDIA Corporation for providing us with the Titan Xp GPU used in this research. This work has been partially funded by the French National Research and Technology Agency (ANRT) and GoPro Technology France.

TABLE III
COMPARISON OF PSNR / ST-RRED ON THE SET8 TESTSET. FOR PNSR: LARGER IS BETTER; BEST RESULTS ARE SHOWN IN BLUE, SECOND BEST IN RED. FOR ST-RRED: SMALLER IS BETTER; BEST RESULTS ARE SHOWN BOLD.

	VNLB	V-BM4D	NV	VNLnet	DVDnet	FastDVDnet
$\sigma = 10$	37.26 / **2.86**	36.05 / 3.87	35.67 / 3.42	37.10 / 3.43	36.08 / 4.16	36.43 / 3.00
$\sigma = 20$	33.72 / **6.28**	32.19 / 9.89	31.69 / 12.48	33.88 / 6.88	33.49 / 7.54	33.37 / 6.65
$\sigma = 30$	31.74 / **11.53**	30.00 / 19.58	28.84 / 33.19	-	31.79 / 12.61	31.60 / 11.85
$\sigma = 40$	30.39 / 18.57	28.48 / 32.82	26.36 / 47.09	30.55 / 19.71	30.55 / 19.05	30.37 / **18.45**
$\sigma = 50$	29.24 / 27.39	27.33 / 49.20	25.46 / 57.44	29.47 / 29.78	29.56 / 27.97	29.42 / **26.75**

TABLE IV
COMPARISON OF PSNR / ST-RRED ON THE DAVIS TESTSET. FOR PNSR: LARGER IS BETTER; BEST RESULTS ARE SHOWN IN BLUE, SECOND BEST IN RED. FOR ST-RRED: SMALLER IS BETTER; BEST RESULTS ARE SHOWN BOLD.

	VNLB	V-BM4D	VNLnet	DVDnet	FastDVDnet
$\sigma = 10$	38.85 / 3.22	37.58 / 4.26	35.83 / **2.81**	38.13 / 4.28	38.97 / 3.49
$\sigma = 20$	35.68 / 6.77	33.88 / 11.02	34.49 / **6.11**	35.70 / 7.54	35.86 / 7.46
$\sigma = 30$	33.73 / 12.08	31.65 / 21.91	-	34.08 / 12.19	34.06 / 13.08
$\sigma = 40$	32.32 / 19.33	30.05 / 36.60	32.32 / 18.63	32.86 / **18.16**	32.80 / 20.39
$\sigma = 50$	31.13 / 28.21	28.80 / 54.82	31.43 / 28.67	31.85 / **25.63**	31.83 / 28.89

TABLE V
Comparison of running times. TIME TO DENOISE A COLOR FRAME OF RESOLUTION 960×540. NOTE: VALUES DISPLAYED FOR VNLB DO NOT INCLUDE THE TIME REQUIRED TO ESTIMATE MOTION.

Method	**Time (s)**
VNLB	420
V-BM4D	156
DVDnet (GPU)	8
VNLnet (GPU)	2.6
FastDVDnet (GPU)	0.1

REFERENCES

[1] Matias Tassano, Julie Delon, and Thomas Veit, "DVDnet: A fast network for deep video denoising," in *IEEE International Conference on Image Processing*, Sep 2019.

[2] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, number 8, pp. 2774–2781.

[3] Yunjin Chen and Thomas Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, Jun 2017.

[4] K Dabov, A Foi, and V Katkovnik, "Image denoising by sparse 3D transformation-domain collaborative filtering," *IEEE Transactions on Image Processing (TIP)*, vol. 16, no. 8, pp. 1–16, 2007.

[5] M. Lebrun, A. Buades, and J. M. Morel, "A nonlocal bayesian image denoising algorithm," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1665–1688, Jan 2013.

[6] H.C. Burger, C.J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2392–2399.

[7] V. Santhanam, V.I. Morariu, and L.S. Davis, "Generalized Deep Image to Image Regression," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo, "Multi-level wavelet-CNN for image restoration," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2018.

[9] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, Jul 2017.

[10] Kai Zhang, Wangmeng Zuo, and Lei Zhang, "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, Sep 2018.

[11] Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand, "Deep joint demosaicking and denoising," *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 1–12, Nov 2016.

[12] Eli Schwartz, Raja Giryes, and Alex M. Bronstein, "DeepISP: Toward learning an end-to-end image processing pipeline," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 912–923, Feb 2019.

[13] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun, "Learning to See in the Dark," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, may 2018, pp. 3291–3300.

[14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation*, vol. 9351 of *Lecture Notes in Computer Science*, chapter chapter 28, pp. 234–241, Springer International Publishing, 2015.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[16] Matias Tassano, Julie Delon, and Thomas Veit, "An analysis and implementation of the ffdnet image denoising method," *Image Processing On Line*, vol. 9, pp. 1–25, Jan 2019.

[17] Anil C Kokaram, *Motion picture restoration: digital algorithms for artefact suppression in degraded motion picture film and video*, Springer Science & Business Media, 1998.

[18] Matteo Maggioni, Giacomo Boracchi, Alessandro Foi, and Karen Egiazarian, "Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms," *IEEE Transactions on Image Processing*, vol. 21, no. 9, pp. 3952–3966, Sep 2012.

[19] Pablo Arias and Jean-Michel Morel, "Video denoising via empirical bayesian estimation of space-time patches," *Journal of Mathematical Imaging and Vision*, vol. 60, no. 1, pp. 70–93, Jan 2018.

[20] Xinyuan Chen, Li Song, and Xiaokang Yang, "Deep rnns for video denoising," Sep 2016, vol. 9971 of *SPIE Proceedings*, p. 99711T, SPIE.

[21] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio, "On the difficulty of training recurrent neural networks," in *ICML*, 2013, pp. 1310–1318.

[22] Thijs Vogels, Fabrice Rousselle, Brian Mcwilliams, Gerhard Röthlin, Alex Harvill, David Adler, Mark Meyer, and Jan Novák, "Denoising with kernel prediction and asymmetric loss functions," *ACM Transactions on Graphics*, vol. 37, no. 4, pp. 1–15, Jul 2018.

[23] Axel Davy, Thibaud Ehret, Gabriele Facciolo, Jean-Michel Morel, and Pablo Arias, "Non-local video denoising by cnn," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[24] Tamara Seybold, *Noise Characteristics and Noise Perception*, pp. 235–265, Springer International Publishing, 2018.

[25] K. Seshadrinathan and A.C. Bovik, "Motion tuned spatio-temporal quality assessment of natural videos," *IEEE Transactions on Image Processing*, vol. 19, no. 2, pp. 335–350, Feb 2010.

[26] Ce Liu and William Freeman, "A high-quality video denoising algorithm based on reliable motion estimation," in *European Conference on Computer Vision (ECCV)*. 2015, pp. 706–719, Springer.
[27] Antoni Buades, Jose-Luis Lisani, and Marko Miladinovic, "Patch-based video denoising with optical flow estimation," *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2573–2586, Jun 2016.
[28] Shangzhe Wu, Jiarui Xu, Yu-Wing Tai, and Chi-Keung Tang, "Deep High Dynamic Range Imaging with Large Foreground Motions," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 117–132.
[29] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox, "Flownet: Learning optical flow with convolutional networks," Dec 2015, pp. 2758–2766, IEEE.
[30] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," Jun 2016, pp. 1874–1883, IEEE.
[31] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems (NIPS)*, pp. 1–9, 2012.
[32] Sergey Ioffe and Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *International Conference on Machine Learning (ICML)*. 2015, pp. 448–456, JMLR.org.
[33] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid, "DeepFlow: Large displacement optical flow with deep matching," in *IEEE International Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013.
[34] Gunnar Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Proceedings of the 13th Scandinavian Conference on Image Analysis*, Berlin, Heidelberg, 2003, SCIA'03, pp. 363–370, Springer-Verlag.
[35] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy, "Liteflownet: A lightweight convolutional neural network for optical flow estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018, pp. 8981–8989.
[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.
[37] Anna Khoreva, Anna Rohrbach, and Bernt Schiele, "Video object segmentation with language referring expressions," in *ACCV*, 2018.
[38] Adam Paszke, Gregory Chanan, Zeming Lin, Sam Gross, Edward Yang, Luca Antiga, and Zachary Devito, "Automatic differentiation in PyTorch," *Advances in Neural Information Processing Systems 30*, pp. 1–4, 2017.
[39] D.P. Kingma and J.L. Ba, "ADAM: a Method for Stochastic Optimization," *Proc. ICLR*, pp. 1–15, 2015.
[40] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna, "Rethinking the Inception Architecture for Computer Vision," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, dec 2015.
[41] Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht, "The marginal value of adaptive gradient methods in machine learning," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 4148–4158.
[42] ABSoft, "Neat Video," https://www.neatvideo.com, 1999–2019.
[43] Rajiv Soundararajan and Alan C. Bovik, "Video quality assessment by reduced reference spatio-temporal entropic differencing," *IEEE Transactions on Circuits and Systems for Video Technology*, 2013.

Matias Tassano received B.Sc. and a M.Sc. degrees in electrical engineering from the UdelaR University, Montevideo, Uruguay. He also received a M.Sc. degree in applied mathematics from the École Normale Supérieure de Cachan, France. He is currently pursuing a Ph.D. degree in applied mathematics with the University of Paris Descartes, Paris, France. His current research interests include machine learning, and image and video processing. In 2015, he was awarded the first place of the Annual National Postgraduate Thesis Contest, Electrical Engineering, by the Uruguayan National Academy of Engineering (ANIU).

Julie Delon is full professor of mathematics at Paris Descartes University, Paris France. She is a member of the laboratoire MAP5, UMR 8145, and she has been elected a member of the Institut Universitaire de France. She is an associate editor for Image Processing on Line (www.ipol.im), the first journal publishing reproducible algorithms, software and online executable articles. Her research interest include stochastic and statistical modeling for image editing and restoration, and numerical optimal transport for imaging and computer vision. In 2018, she received the Blaise Pascal award from the French Académie des Sciences.

Thomas Veit received a Ph.D. in computer vision from Inria-Université de Rennes in 2005. He worked as a research scientist on driving assistance system from 2007 to 2011 at the French National Institute for Road Safety (INRETS). In 2011, he joined DxO Labs to focus on digital cameras and image quality. Since 2015, he is with GoPro enhancing video quality and camera features.