Video joint denoising and demosaicing with recurrent CNNs
Supplementary material

Valéry Dewil[†] Adrien Courtois[†] Mariano Rodríguez[†] Thibaud Ehret[†]
Nicola Brandonisio[*] Denis Bujoreanu[*] Gabriele Facciolo[†] Pablo Arias[†]

[†] Université Paris-Saclay, CNRS, ENS Paris-Saclay, Centre Borelli, 91190, Gif-sur-Yvette, France
[*] Huawei Technologies France SASU
https://centreborelli.github.io/RVDD/

1. Network architecture

We consider a U-Net architecture shown in Figure 1, as it is simple and due to its multiscale nature it provides a good trade-off between denoising quality and computational cost. Our U-Net has the following characteristics:

- U-Net with 4 dyadic scales
- fusion of skip connections via concatenation
- two convolution layers in each scale of encoder/decoder paths
- upscaling using bilinear upsampling followed by convolution
- downscaling using a convolution followed by max-pooling
- all convolutions are 2D convolutions with 3x3 filters and output feature maps of 48 channels
- inputs: 2 packed raw frames concatenated together as a 8 channel tensor of size $W/2 \times H/2$ (with optionally an occlusion mask as a 9th channel)
- outputs: 1 packed raw frame (4 channels tensor of size $W/2 \times H/2$).

The architecture based on ConvNeXt U-Net (see diagram in Figure 2) provides better results than the standard U-Net (see PSNR/SSIM results in Table 4 of the main paper and Table 3 of this supplementary material). For the baseline (RVDD-basic), the gain is marginal, however the ConvNeXt U-Net converges much faster. In Figure 3, we show a plot of the PSNR obtained in our validation dataset for each epoch and for both ISO. The ConvNeXt U-Net achieves the convergence from about epoch 22 while the standard one needs 100 epochs to converge. In addition to converging faster, with the full RVDD method it achieves a higher performance (the gain in PSNR is 0.2dB for the ISO 3200 and 0.3dB for the ISO 12800).

2. Training loss

The loss of our recurrent network with T unrollings is a weighted sum of T individual L1 losses that are computed with the denoised frame for each unrolling. We recall that the output of the network is computed as

$$\widehat{u}_t = \mathcal{F}\left(\mathcal{W}_{t-1,t}\varphi^L_{t-1}, \mathcal{W}_{t-1,t}u_{t-1}, ... \right. \\ \left. \mathcal{D}(f_t), \mathcal{W}_{t+1,t}\mathcal{D}(f_{t+1})\right), \quad (1)$$

where f_t and f_{t+1} are two raw noisy frame, $\mathcal{D}$ is a demosaicing operator, $\mathcal{W}_{t-1\to t}$ and $\mathcal{W}_{t+1\to t}$ are two warping operators to compensate for motion (defined in Equation 3 from the main paper) and φ^L_{t-1} is the feature map from the last hidden layer (see Section 3 from the main paper for more details). When training, we run the network on short videos of $T+1$ frames (or $T+2$ if we are using the future frame) to generate T output frames $\widehat{u}_1, ..., \widehat{u}_T$. For the first output $\widehat{u}_1$ the previous feature map φ^L_0 is initialized as zero, and the previous output $\widehat{u}_0$ as the previous noisy raw frame f_0. The loss is computed by

$$\text{loss}\left((\widehat{u}_t)_{t=1,...,T}, (u_t)_{t=1,...,T}\right) = \sum_{t=1}^{T} \lambda_t \left\|\widehat{u}_t - u_t\right\|_1, \quad (2)$$

where the weights λ_t are non-negative and sum to one. The weights control the importance given to each output. We vary the weights during training. For the first 20 epochs, we only train the first unrolling by setting all the weight on the first output, i.e. $\lambda_1 = 1$ and $\lambda_t = 0$ for $t \geq 1$. This is mainly to speed up the training, as we only need to compute the first unrolling. Starting at epoch 20 to 25, we gradually shift the weights until 90% of the weight is given to the last

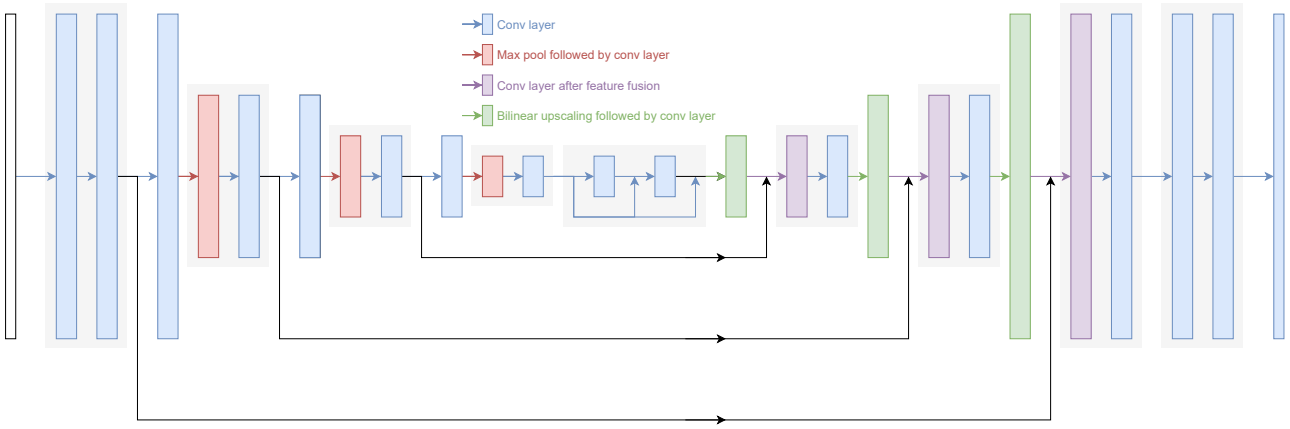

Figure 1: Network diagram of the U-Net.

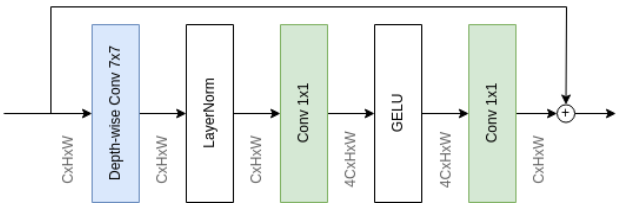

Figure 2: Structure of the ConvNeXt block [5].

	φ^L_{t-1}	f_{t+1}	Lin. RGB PSNR 3.2k	Lin. RGB PSNR 12.8k	Lin. RGB SSIM 3.2k	Lin. RGB SSIM 12.8k
RVDD-basic	✗	✗	44.74	40.73	0.989	0.977
	✓	✗	44.99	41.05	0.989	0.979
	✗	✓	45.05	41.14	0.989	0.979
RVDD	✓	✓	**45.29**	**41.45**	**0.990**	**0.981**

Table 1: PSNR and SSIM in the linear RGB domain for the different frameworks for handling the recurrence (see Section 2 in the main paper) in the validation set of our synthetic dataset. We consider two ISO levels taken from the CRVD dataset. Best results are in **bold**.

unrolling and the remaining 10% is split uniformly between the first $T-1$ unrollings, i.e. $\lambda_t = \frac{1}{10(T-1)}, t = 1, ..., T-1$ and $\lambda_T = \frac{9}{10}$. The rationale for these weights is to give more importance to the last unrolling, as it is the one more similar to the steady state of the networks operation in a video, while still giving some weight to the first unrollings, as they are necessary to reach that steady state.

3. Quantitative results on the linear RGB domain

In the main paper, we reported the PSNR and SSIM values on average in the validation set and in the sRGB domain (after a post-processing pipeline). In this section, we report the PSNR and SSIM values in the linear RGB domain (no post-processing). Table 1 shows the effect of the different inputs to our RVDD network on our dataset with the two ISO levels. Recall that RVDD-basic denotes the network with only two inputs: the current noisy frame f_t and the previous RGB output $\hat{u}_{t-1}$, whilst RVDD (the full configuration) includes the features from the previous frame φ^L_{t-1} and the future frame f_{t+1}. In Table 2, we compare our method with the FastDVDnet-JDD described in the main paper. In Table 3, we compare the standard U-Net with the ConvNeXt U-Net.

4. Visual results on real data

In this section, we present the results obtained by applying RVDD with the ConvNeXt U-Net on the outdoor sequences of the CRVD [7] dataset. We compare against two methods: FastDVDnet-JDD and Multi-Frame-to-Frame (MF2F) [1]. In [1], the authors proposed a self-supervised framework for fine-tuning a pre-trained denoising network to a new noise type. They achieve joint denoising and demosaicing by demosaicing the noisy raw images (using [4]) and then fine-tuning a FastDVDnet on the demosaiced raw (initially trained for handling additive white Gaussian noise). The results are shown in Figure 4. Videos of noisy sequences and of results obtained with the different methods are attached to the supplementary material. RVDD recovers more details than FastDVDnet-JDD. Globally it has a better reconstruction of the textures.

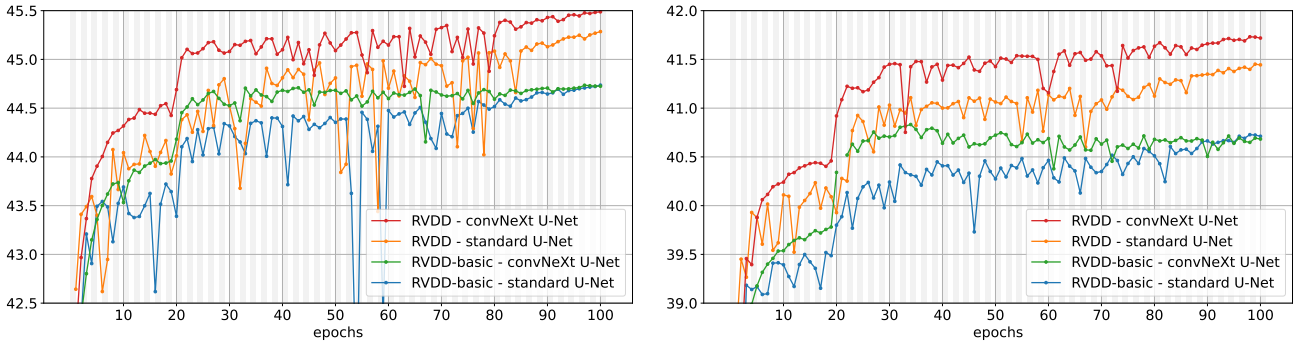

Figure 3: Evolution of validation PSNR during training of our RVDD and RVDD-basic models with the standard U-Net and the convNeXt U-Net. On the left, ISO 3200 and right ISO 12800.

network	$\mathcal{W}$	f_{t+1}	trained on	non-stabilized				stabilized			
				Lin. RGB PSNR		Lin. RGB SSIM		Lin. RGB PSNR		Lin. RGB SSIM	
				3.2k	12.8k	3.2k	12.8k	3.2k	12.8k	3.2k	12.8k
FastDVDnet-JDD			non stab.	43.06	38.35	0.983	0.963	43.99	39.51	0.986	0.970
VDD	✗	✗	non stab.	43.18	38.88	0.984	0.967	43.60	39.36	0.985	0.970
VDD	✗	✓	non stab.	43.18	38.89	0.984	0.966	43.83	39.63	0.986	0.971
VDD	✓	✗	non stab.	44.04	39.88	0.986	0.973	44.35	40.09	0.987	0.974
VDD	✓	✓	non stab.	44.56	40.55	0.988	0.976	44.88	40.77	0.989	0.977
RVDD-basic	✓	✗	non stab.	44.74	40.73	0.989	0.977	45.09	40.97	0.990	0.979
RVDD	✓	✓	non stab.	**45.29**	**41.45**	**0.990**	**0.981**	**45.56**	**41.67**	**0.991**	**0.982**
FastDVDnet-JDD			stab.	42.86	38.40	0.982	0.963	44.18	40.20	0.986	0.974
VDD	✗	✗	stab	43.03	38.78	0.983	0.966	44.08	39.80	0.986	0.972
VDD	✗	✓	stab	42.93	38.57	0.983	0.964	44.23	40.04	0.987	0.973
VDD	✓	✗	stab	43.97	39.81	0.986	0.972	44.43	40.13	0.988	0.974
VDD	✓	✓	stab	44.51	40.49	0.988	0.976	45.01	40.85	0.989	0.978
RVDD-basic	✓	✗	stab	44.66	40.72	0.989	0.978	45.19	41.12	0.990	0.980
RVDD	✓	✓	stab	**45.14**	**41.33**	**0.990**	**0.980**	**45.70**	**41.76**	**0.991**	**0.982**

Table 2: PSNR and SSIM in the linear RGB domain in the validation set of our synthetic dataset. We compare our JDD adaptation of FastDVDnet [6] with six variants of our network: the two frame recurrent RVDD, RVDD-basic and four non-recurrent networks labeled VDD: with/without warping ($\mathcal{W}$) and with/without the future frame f_{t+1}.

Architecture	Lin. RGB PSNR		Lin. RGB SSIM	
	3.2k	12.8k	3.2k	12.8k
RVDD-basic U-Net	44.74	40.73	0.989	0.977
RVDD-basic ConvNeXt U-Net	44.73	40.83	0.989	0.977
RVDD U-Net	45.29	41.45	**0.990**	0.981
RVDD ConvNeXt U-Net	**45.49**	**41.73**	**0.990**	**0.982**

Table 3: PSNR and SSIM in the linear RGB domain for RVDD using the standard U-Net and our improved version with ConvNeXt blocks in the validation set of our synthetic dataset. We consider two ISO levels taken from the CRVD dataset. Best results are in **bold**.

5. Modified version of FastDVDnet for JDD

In the main paper, we adapted FastDVDnet [6] for handling the JDD task. We proposed a simple adaptation of FastDVDnet by demosaicing the frames before feeding the network. This version corresponds to an *early* demosaicing approach (see Figure 5(a)). We also tested another adaptation in which we applied a *late* demosaicing. For this modified the input layer of the first U-Net so that it takes mosaiced frames packed in four channels at half-resolution. At the final layer of the first U-Net, a twelve-channel image is produced and then upscaled with a non-trainable upsampling (*pixel shuffle*) into a three-channels image. In order to apply the skip connection at the original scale, the middle frame of the input temporal window is demosaiced using the Hamilton-Adams demosaicing [3, 2]. The second U-Net then takes three-channel frames and outputs a three-channel frame as in the early demosaicing version. This modified architecture is trained with the same hyperpame-

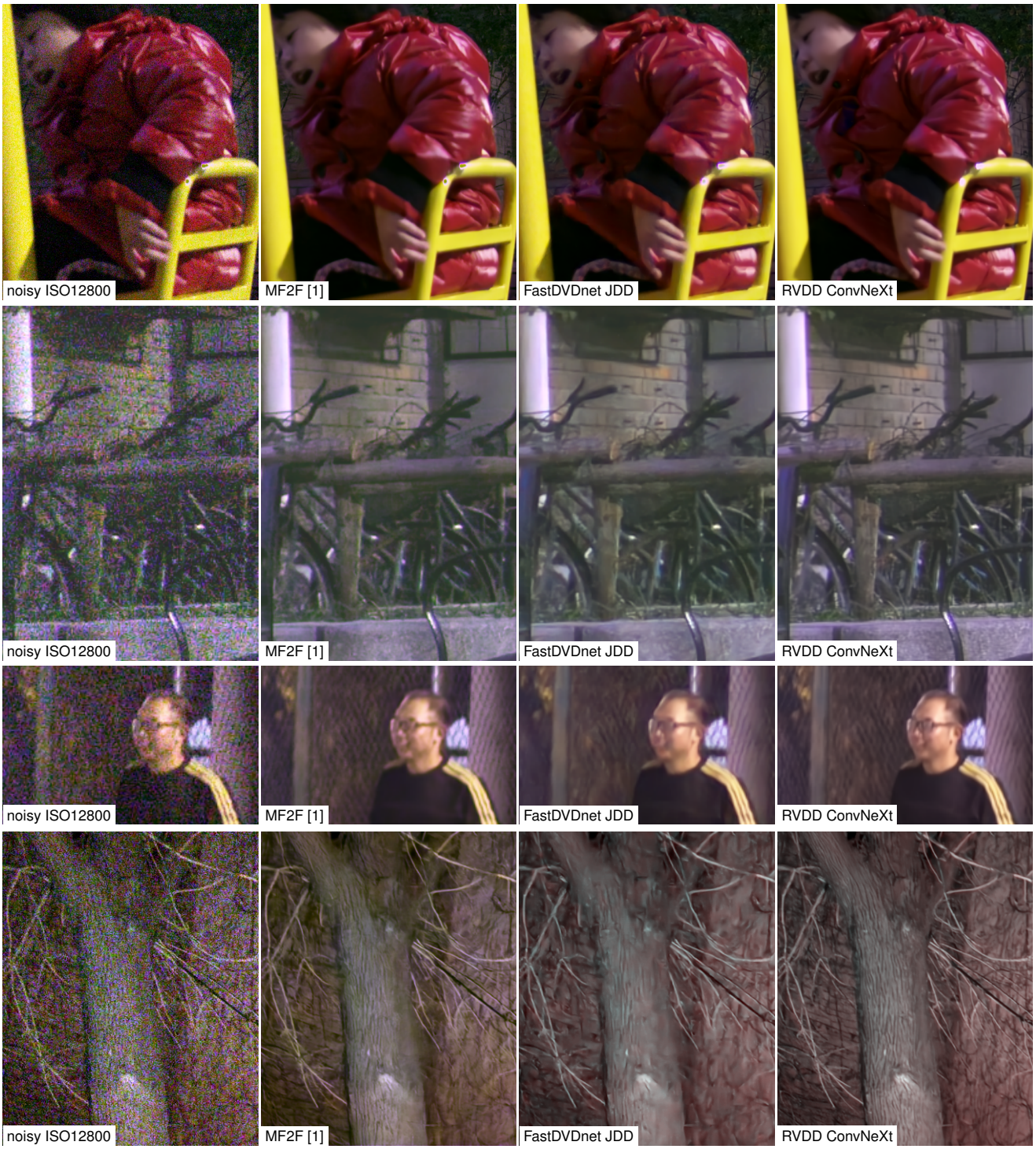

Figure 4: Results obtained with our joint denoising and demosaicing method (RVDD) on real raw videos from the CRVD dataset [7]. For comparison we show results obtained with the self-supervised video denoising method MF2F [1] and with an adaptation of FastDVDnet [6] to JDD.

ters as the first version (early demosaicing) presented in the main paper, except the patch size which is doubled for the late demosaicing so that the first U-Net of both adaptations work at the same resolution. In Figure 5(b), we show a dia-

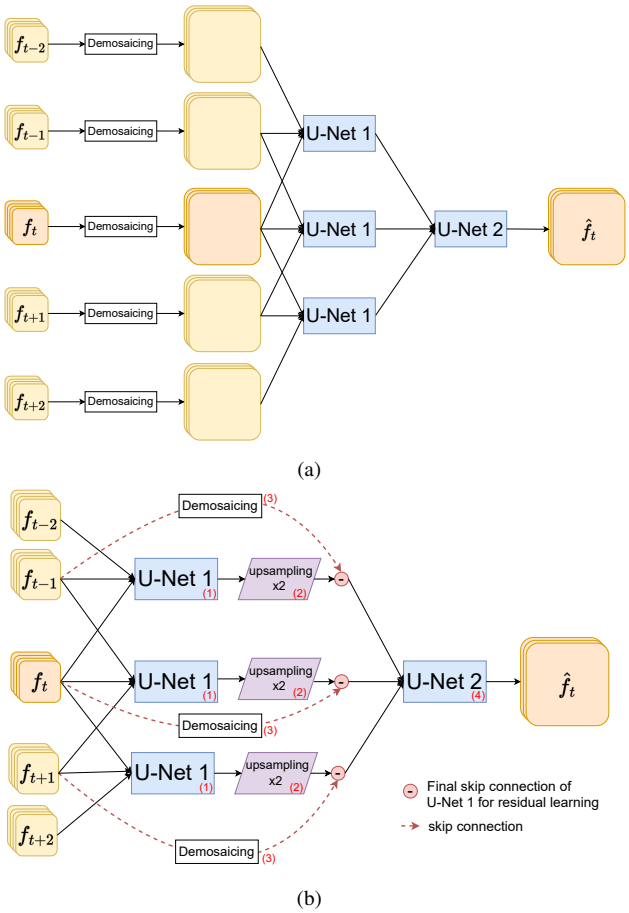

Figure 5: Modified architectures of FastDVDnet [6] for performing joint denoising and demosaicing. (a) First version (called *early* demosaicing) : the raw input packed in 4 channels of half-resolution are demosaiced using the Hamilton-Adams demosaicing [3, 2], then U-Net 1 and 2 are applied on RGB images as in the original FastDVDnet [6]. (b) Second version (called *late* demosaicing): U-Net 1 takes a temporal window of three contiguous raw frames packed in 4 channels (1), U-Net 1 is followed by an non-trainable upsampling layer (2) which produces 3 channel images (*pixel shuffling*), the 4 channels input frame is demosaiced using the Hamilton-Adams demosaicing [3, 2] (3) for the final skip connection. This is repeated for the three possible windows of three contiguous frames and the three outputs are used as input for the U-Net 2 which produces the denoised result (4).

gram of the late demosaicing adaptation of FastDVDnet for JDD.

Both version, late and early demosaicing, attain a very similar performances. The early demosaicing (explained in the main paper) has a slightly higher performance, but the late demosaicking approach offers a lighter alternative.

References

[1] Valéry Dewil, Jérémy Anger, Axel Davy, Thibaud Ehret, Gabriele Facciolo, and Pablo Arias. Self-supervised training for blind multi-frame video denoising. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2724–2734, 2021.

[2] Qiyu Jin, Yu Guo, Jean-Michel Morel, and Gabriele Facciolo. A mathematical analysis and implementation of residual interpolation demosaicking algorithms. *Image Processing On Line*, 11:234–283, 2021.

[3] James E. Adams Jr. and John F. Hamilton Jr. Adaptive color plane interpolation in single sensor color electronic camera, US Patent 5,629,734, Nov. 1996.

[4] Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Minimized-laplacian residual interpolation for color image demosaicking. In *Digital Photography X*, volume 9023, page 90230L. International Society for Optics and Photonics, 2014.

[5] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[6] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In *The IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1354–1363, June 2020.

[7] Huanjing Yue, Cong Cao, Lei Liao, Ronghe Chu, and Jingyu Yang. Supervised raw video denoising with a benchmark dataset on dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2301–2310, 2020.