Video Noise Removal Using Progressive Decomposition With Conditional Invertibility

Haoran Huang[1] Yuhui Quan[1,2] Zhenghua Lei[1] Jinlong Hu[1] Yan Huang[1,2,*]
[1] South China University of Technology, Guangzhou 510006, China
[2] Pazhou Lab, Guangzhou 510335, China
csherry@mail.scut.edu.cn, csyhquan@scut.edu.cn, zhenghua.lei.scut@gmail.com,
jlhu@scut.edu.cn, aihuangy@gmail.com

***Abstract*—Video denoising aims at removing noise from noisy video frames and meanwhile preserving their structures and details. It is a challenging task, as both noise and video structures/details correspond to high-frequency components of a noisy video which are hard to distinguish. This paper proposes a deep video denoiser using a progressive decomposition process with conditional invertibility. Noisy video frames are first decomposed into two latent codes via a forward process of conditional invertible coupling layers, where one latent code carries the maximal information regarding the noise-free reference frame while the other encodes the information regarding noise, misalignment and content difference. The clean video is then reconstructed from the latent codes of noise-free frames using the reverse pass of the coupling layers. To improve the robustness to variant noise levels, the coupling layers are conditioned on noise level. In addition, memory units are introduced to the conditioned coupling layers to better exploit temporal correlation among frames for feature disentanglement. Experiments on two benchmark datasets have demonstrated the effectiveness of our method.**

***Index Terms*—Video Denoising, Video Restoration, Coupling Layers, Feature Disentanglement**

I. Introduction

Video is a prevailing digital media for recording and disseminating information in daily life. Noise corruption is inevitable during video acquisition, transmission and storage, causing noticeable degradation of visual quality of a video. Video denoising is to remove noise from a video, which is an important step in video processing and valuable to the fields of digital photography and social multimedia [1], [2].

Video denoising can be naively done by applying an image denoiser (*e.g.* [3]–[8]) to each frame individually. However, this cannot make full use of the temporal information along adjacent frames. A dominant approach is using multiple consecutive noisy frames to restore the reference frame, which can be done using patch-based methods [9]–[12] or motion compensation (MC)-based methods [1], [13], [14]. Patch-based methods exploit the recurrence of spatio-temporal patches and denoise them by a handcrafted process or a learned deep neural network (DNN). For instance, to denoise a group of similar spatio-temporal patches, V-BM4D [10] uses a joint transform and thresholding process, VNLB [9] utilizes an empirical Bayes estimator, and VNLNet [11] leverages a simplified DnCNN [3]. In general, patch-based methods need to search similar counterparts within a volumetric neighborhood for each noisy patch for denoising and stack the denoised patches back to obtain a clean image. This often results in inconsistency on overlapping patch areas and is time-consuming.

Different from patch-based methods, MC-based methods operate on adjacent frames aligned by MC. For instance, DVDNet [13] uses optical flow to warp adjacent frames to the reference frame and then applies spatio-temporal deep denoising. FastDVDNet [14] improves the speed of DVDNet by avoiding explicit optical flow prediction. MMNet [1] simultaneously recovers multiple clean frames from consecutive noisy frames. ER2R [15] combined the image-based R2R loss [16], [17] with MC for self-supervised video denoising. The performance of MC-based methods can be noticeably affected by the misalignment caused by MC estimation error or content difference. For improvement, PaCNet [18] incorporates patch matching into DNN-based video denoisers to merry the advantages of the both.

How to remove noise while keeping original video details is a key problem for video denoising. Since both random noise and video structures (*e.g.* textures and edges) are the high-frequency parts of a noisy video, it is difficult to distinguish noise from various video structures. Currently, there is still much room for improvement in existing methods. In this paper, motivated by existing invertible image processing techniques [7], [19], we interpret video denoising as a progressive decomposition process with conditional invertibility. Initially-aligned adjacent video frames are passed through a series of conditional invertible coupling layers [20] in a forward process, by which the frames are progressively decoupled into two latent representations: one encoding the maximal information shared by latent clean video frames, and the other encoding the information of noise and misaligned content. The noise-free video is then reconstructed from the latent representation associated with latent clean video frames, using a reverse pass of the coupling layers.

There are two branches in each conditional coupling layer, respectively responsible for extracting common features and noise/misaligned content. Using such coupling layers as the

This work was supported by Natural Science Foundation of Guangdong Province (Grant No. 2022A1515011755 and 2023A1515012841) and Science and Technology Plan Project of Guangzhou (Grant No. 2023A04J1681).

*Corresponding author: Yan Huang

basic blocks not only ensures the information-losslessness of feature extraction to benefit the reconstruction process, but also enables rich interactions between the two branches for feature disentanglement. In addition, we incorporate the convolutional Long short-term memory (ConvLSTM) [21] into the coupling layers to better capture temporal information in the deep feature domain, and we also condition the coupling layers on a noise level map estimated by a sub-DNN, so as to improve the model's ability in handling videos with variant noise levels.

To summarize, this paper proposes a deep learning-based video denoising method. Its contributions are listed as follows:

- A DNN for video denoising is proposed with a progressive decomposition framework with conditional invertibility. It separates the noise and misalignment information from clean video features in the forward process and reconstructs the clean video in the reverse process, with an information-losslessness property.
- ConvLSTM is incorporated into the coupling layers to further capture temporal information for better feature disentanglement during the progressive decomposition.
- Noise level maps are incorporated into the coupling layers as a conditional input, enabling a single learned DNN model to handle videos with variant noise levels.

II. Proposed Method

A. Preliminary on Coupling Layers

Our proposed DNN is built upon invertible coupling layers [20]. For better understanding, we first briefly introduce coupling layers. A coupling layer provides a specific double-branch form of the forward process for converting input to output so that it can replicate the input simply through its reverse mode. Such an invertibility allows the processing to be information-lossless. As shown in Fig. 1, a coupling layer first divide its input $\boldsymbol{X}$ into two parts $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ at channel dimension. Then $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are transformed and interacted with each other by some function set $\{\phi_1, \phi_2, \psi_1, \psi_2\}$ in a coupling way, leading to $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$ which are finally concatenated as $\boldsymbol{Y}$. Such steps form the forward process that can be expressed as follows:

$$\boldsymbol{X}_1, \boldsymbol{X}_2 = \text{split}(\boldsymbol{X}), \tag{1}$$

$$\boldsymbol{Y}_1 = \boldsymbol{X}_1 \odot \exp(\phi_1(\boldsymbol{X}_2)) + \psi_1(\boldsymbol{X}_2), \tag{2}$$

$$\boldsymbol{Y}_2 = \boldsymbol{X}_2 \odot \exp(\phi_2(\boldsymbol{Y}_1)) + \psi_2(\boldsymbol{Y}_1), \tag{3}$$

$$\boldsymbol{Y} = \text{concat}(\boldsymbol{Y}_1, \boldsymbol{Y}_2), \tag{4}$$

where $\odot$ denotes element-wise multiplication. The interactive functions $\phi_1, \phi_2, \psi_1, \psi_2$ are implemented by arbitrary DNN sub-networks and their definitions do not affect the invertibility of the coupling layer. The reverse process of the coupling layer can then be simply done as follows:

$$\boldsymbol{Y}_1, \boldsymbol{Y}_2 = \text{split}(\boldsymbol{Y}), \tag{5}$$

$$\boldsymbol{X}_2 = (\boldsymbol{Y}_2 - \psi_2(\boldsymbol{Y}_1)) \oslash \exp(\phi_2(\boldsymbol{Y}_1)), \tag{6}$$

$$\boldsymbol{X}_1 = (\boldsymbol{Y}_1 - \psi_1(\boldsymbol{X}_2)) \oslash \exp(\phi_1(\boldsymbol{X}_2)), \tag{7}$$

$$\boldsymbol{X} = \text{concat}(\boldsymbol{X}_1, \boldsymbol{X}_2), \tag{8}$$

where $\oslash$ denotes element-wise division.

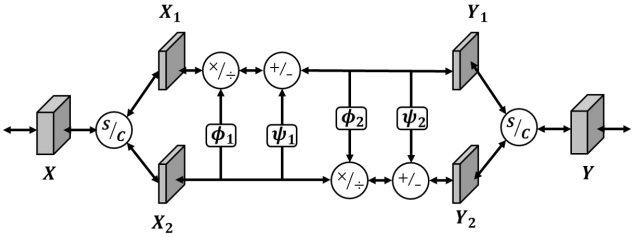

Fig. 1. Structure of a standard coupling layer.

B. DNN Architecture

The proposed DNN for video denoising is called IVDNet (Invertible Video Denoising Network) and outlined in Fig. 2. It aims at obtaining a denoised reference frame $\boldsymbol{F}_d^t \in \mathbb{R}^{W\times H\times 3}$ from given consecutive noisy frames $\{\boldsymbol{F}_n^i\}_{i=t-m}^{i=t+m} \in \mathbb{R}^{W\times H\times 3}$, where m is the number of previous/subsequent adjacent frames which is set 2 in our practice. The IVDNet consists of a frame alignment block (FAB), a motion refinement block (MRB), a noise estimation block (NEB), and several conditional coupling layers (CCLs). In inference, the IVDNet performs a forward pass for the disentanglement of noise/image features and then calls a reverse pass for noise-free frame reconstruction.

In the forward process, the input frames first sequentially pass through the FAB and the MRB for initial frame alignment and motion refinement, respectively. Then, the aligned frames are down-sampled by pixel unshuffle and divided into two parts as the input of the subsequent double-branch coupling layers. Finally, the IVDNet extracts and decouples features through a series of coupling layers conditioned on the noise level map estimated by the NEB, and then it outputs the latent feature map $\boldsymbol{y}^t$ and noise latent code $\boldsymbol{z}$. The whole forward process can be expressed as follows:

$$\text{CCL}_{\times k}^{\boldsymbol{N}}(\text{MRB}(\text{FAB}(\{\boldsymbol{F}_n^i\}_{i=t-m}^{i=t+m}))) \rightarrow \boldsymbol{y}^t, \boldsymbol{z}, \tag{9}$$

where $\boldsymbol{N} \in \mathbb{R}^{H\times W}$ denotes a spatially-variant noise map estimated by the NEB, and $\text{CCL}_{\times k}^{\boldsymbol{N}}(\cdot)$ denotes k sequentially-connected CCLs conditioned by $\boldsymbol{N}$, with k set to 8 in practice.

Using the progressive decomposition in the forward process, the maximal information shared by adjacent frames (which are likely to correspond to clean video frames) are encoded in the latent feature $\boldsymbol{y}^t$, while the information of noise, misalignment and content difference is encoded in the latent feature $\boldsymbol{z}$. To reconstruct a clean video in the reverse pass, we first cancel the effect of the latent code by zeroing it during inference but replacing it with a random variable $\hat{\boldsymbol{z}}$ sampled from a Gaussian distribution during training for data augmentation and noise-injection-based regularization. Then, we send the latent representation $\boldsymbol{y}^t$ and $\hat{\boldsymbol{z}}$ back to the same coupling layers in their reverse mode, and concatenate the outputs with pixel shuffle upsampling. Finally, the reconstructed frames $\{\boldsymbol{F}_d^i\}_{i=t-m}^{i=t+m}$ with noise removal and full alignment to the reference frame are obtained. We average these denoised frames to obtain the final

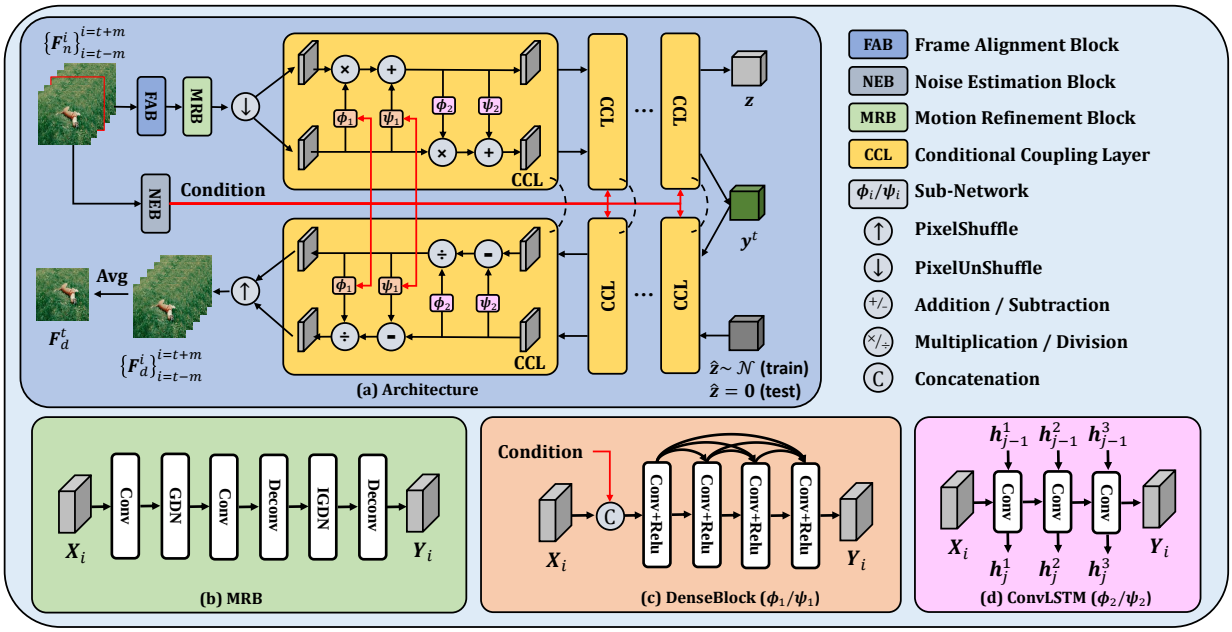

Fig. 2. Outline of proposed IVDNet for video noise removal.

denoising result of the current (reference) frame. Accordingly, the reverse process can be expressed as follows:

$$\text{Avg}(\text{CCL}_{\times k}^{-1}([\boldsymbol{y}^t, \hat{\boldsymbol{z}}])) \rightarrow \boldsymbol{F}_d^t, \tag{10}$$

where Avg denotes averaging along the frame dimension, and CCL^{-1} denotes the CCL in its reverse mode.

C. Frame Alignment and Motion Refinement

Compared to image denoising, video denoising can utilize the similar information of adjacent frames to assist the restoration of the current reference frame. However, due to the motion of objects and the jitter of the capture device, video frames are often not well aligned. To better make use of spatio-temporal information among adjacent frames, we borrow the idea from SpyNet [22], a lightweight flow-based DNN, to construct the FAB for frame alignment in our IVDNet, which calculates the optical flow by combining the classical spatial pyramid with deep learning, so as to achieve a good balance between accuracy and speed. In addition, the MRB with its structure shown in Fig. 2 (b) is introduced to further refine motion vectors for better alignment. It consists of two convolutional (Conv) layers and one GDN [23] followed by two deconvolutional (Deconv) layers and one IGDN [23]. Briefly, the MRB encodes the corrupted motion vector into a compact representation, and then obtains the refined motion vector by a decoding process.

With the FAB and MRB modules, noisy frames are almost aligned in the feature domain, but there will inevitably exist mis-compensated areas due to estimation errors or content differences. Such misaligned information is handled by the subsequent progressive decomposition process with the CCLs described in the next subsection.

D. Conditional Coupling Layer

Each CCL has two branches, and the aligned frames are divided into two parts as its input; see Fig. 2(a). The two branches are responsible for extracting common features and noise/misaligned content information, respectively. Using invertible coupling layers not only ensures the information-losslessness of feature extraction to benefit the reconstruction process, but also enables rich interactions between the two branches for feature disentanglement. To better extract deep features and temporal information, a five-layer DenseBlock (see Fig. 2(c)) and ConvLSTM (see Fig. 2(d)) are selected as the basic functions ϕ_1/ψ_1 and $\phi_2/\ \psi_2$ in the coupling layers respectively. Furthermore, the noise level maps estimated by NEB are incorporated into the CCLs as conditional input, enabling the model to handle videos with various noise levels.

Noise estimation Noise level is an important prior for video denoising. To handle videos with different noise levels, a common solution is to learn a specific model for each noise level, thus requiring multiple models. By incorporating a noise level map into coupling layers as the conditional input, our IVDNet can handle videos with various noise levels using a single model. We define the NEB as the noise level estimator of [24], which utilizes the low-rank characteristic of non-local similar blocks and the eigenvalues of their covariance matrix to estimate the noise level. Interested readers are referred to [24].

ConvLSTM Temporal information existent in neighboring frames can provide additional information to boost the video denoising performance. The ConvLSTM [21] is good at processing temporal information. We incorporate a simplified ConvLSTM module (see Fig. 2(d)) into CCLs for better utilizing the temporal dependencies in adjacent video frames in the deep feature domain to improve prediction.

E. Loss Function

For the t^{th} noisy frame, let $\boldsymbol{F}_c^t, \boldsymbol{F}_d^t$ denote the corresponding ground-truth clean frame and the denoised frame, respectively. The overall training loss function $\mathcal{L}$ consists of an ℓ_1-reconstruction loss and a perceptual loss:

$$\mathcal{L} = \sum_t \|\boldsymbol{F}_d^t - \boldsymbol{F}_c^t\|_1 + \lambda \sum_t \|\Phi(\boldsymbol{F}_d^t) - \Phi(\boldsymbol{F}_c^t)\|_2, \quad (11)$$

where Φ is a set of VGG-16 layers [25], and λ is set to 0.5 in our practice. Such losses are common for image denoising. The loss function indeed encourages the latent code of noise-free frames to reconstruct a noise-free frame. Together with the fact that the CCLs are invertible without information loss nor information gain, it is expected the latent codes can perform feature disentanglement effectively.

III. EXPERIMENTS

The experiments for performance evaluation are conducted on two widely-adopted benchmark datasets: DAVIS [26] and Set8 [13]. Following the experimental configurations of [13], [14], we train our IVDNet on the DAVIS training set and evaluate it on the DAVIS test set and Set8, respectively. Six video denoising methods are used for performance comparison, including VNLB [9], V-BM4D [10], VNLNet [11], DVDNet [13], FastDVDNet [14] and PaCNet [18].

Implementation details The training set consists of multiple input-output pairs, generated by adding Gaussian white noise with noise levels $\sigma \in [5, 50]$ (randomly drawn from a uniform distribution) to cropped clean patches of size 128×128. The test set is generated by adding Gaussian white noise of noise level $\sigma = 10, 20, 30, 40, 50$ to clean frames, respectively. The model weights are initialized by the Xavier [27] method. The Adam [28] optimizer is called with a learning rate of 1e-4 for the first half epochs and 1e-5 for the second half. The batch size is set to 1. The implementation is based on PyTorch and run on a single NVIDIA GeForce GTX 2070Super GPU. The code will be made public on our GitHub.

A. Results and Analysis

Quantitative comparison in performance and complexity The quantitative results on two test datasets in terms of Peak-Signal-to-Noise Ratio (PSNR) are listed in Table I and Table II respectively for comparison. It can be seen that our proposed IVDNet achieved the overall best results both on the DAVIS and Set8 datasets, demonstrating its effectiveness. Particularly, IVDNet is the best performer on four out of five noise levels.

The number of floating-point operations (FLOPS) of our IVDNet is also compared to its top competitor PaCNet, which is 394G over 1340G. In other words, our model also has its advantage in terms of model complexity due to the use of coupling layers, whose number of FLOPS is around only $1/3$ of that of the PaCNet.

Qualitative analysis comparison See Fig. 3 for the output features of several CCLs. Compared to the features extracted for noise-free frames, the features extracted for noise components contain significantly heavier noise. In addition, features for noise-free frames contain less and less noise along the network depth, while the ones for noise components contain more and more noise. This demonstrated that our IVDNet can effectively perform progressive feature disentanglement.

TABLE I
PSNR(DB) RESULTS ON DAVIS TEST SET. BEST (SECOND-BEST) RESULTS OF EACH COLUMN ARE **BOLDFACED** (UNDERLINED).

Method	σ=10	σ=20	σ=30	σ=40	σ=50	Average
VNLB	38.85	35.68	33.73	32.32	31.13	34.34
V-BM4D	37.58	33.88	31.65	30.05	28.80	32.39
VNLNet	35.83	34.49	-	32.32	31.43	-
DVDNet	38.13	35.70	34.08	32.86	31.85	34.52
FastDVDNet	38.71	35.77	34.04	32.82	31.86	34.64
PaCNet	**39.97**	**36.82**	34.79	33.34	32.20	35.42
IVDNet	39.88	**36.82**	**34.96**	**33.58**	**32.47**	**35.52**

TABLE II
PSNR(DB) RESULTS ON SET8. BEST (SECOND-BEST) RESULTS OF EACH COLUMN ARE **BOLDFACED** (UNDERLINED).

Method	σ=10	σ=20	σ=30	σ=40	σ=50	Average
VNLB	**37.26**	33.72	31.74	30.39	29.24	32.47
V-BM4D	36.05	32.19	30.00	28.48	27.33	30.81
VNLNet	37.10	33.88	-	30.55	29.47	-
DVDNet	36.08	33.49	31.79	30.55	29.56	32.29
FastDVDNet	36.44	33.43	31.68	30.46	29.53	32.31
PaCNet	37.06	33.94	32.05	30.70	29.66	32.68
IVDNet	36.62	**33.95**	**32.27**	**31.04**	**30.06**	**32.79**

The qualitative evaluation is done by visually comparing the denoised frames of different methods in Fig. 4. It can be seen that our proposed IVDNet has advantages over other methods to eliminate noise and maintain details. For instance, our results can better preserve the detailed features of faces and hands in the first sample than other compared methods. The superiority of our qualitative results has verified that the proposed progressive decomposition with conditional invertible coupling layers do benefit the reconstruction process.

Fig. 3. Visualization of the features associated to latent image representation $\mathbf{y}^t$ and to noise component $\mathbf{z}$ respectively, output by CCLs. The noise level of the input image is 50. The latent features w.r.t. $\mathbf{y}^t$ and the associated noise components are of 12 and 48 channels respectively, and we show the first three channels as RGB images.

B. Ablation Studies

To analyze the contribution of each key component in our proposed IVDNet, we conduct ablation studies by forming the

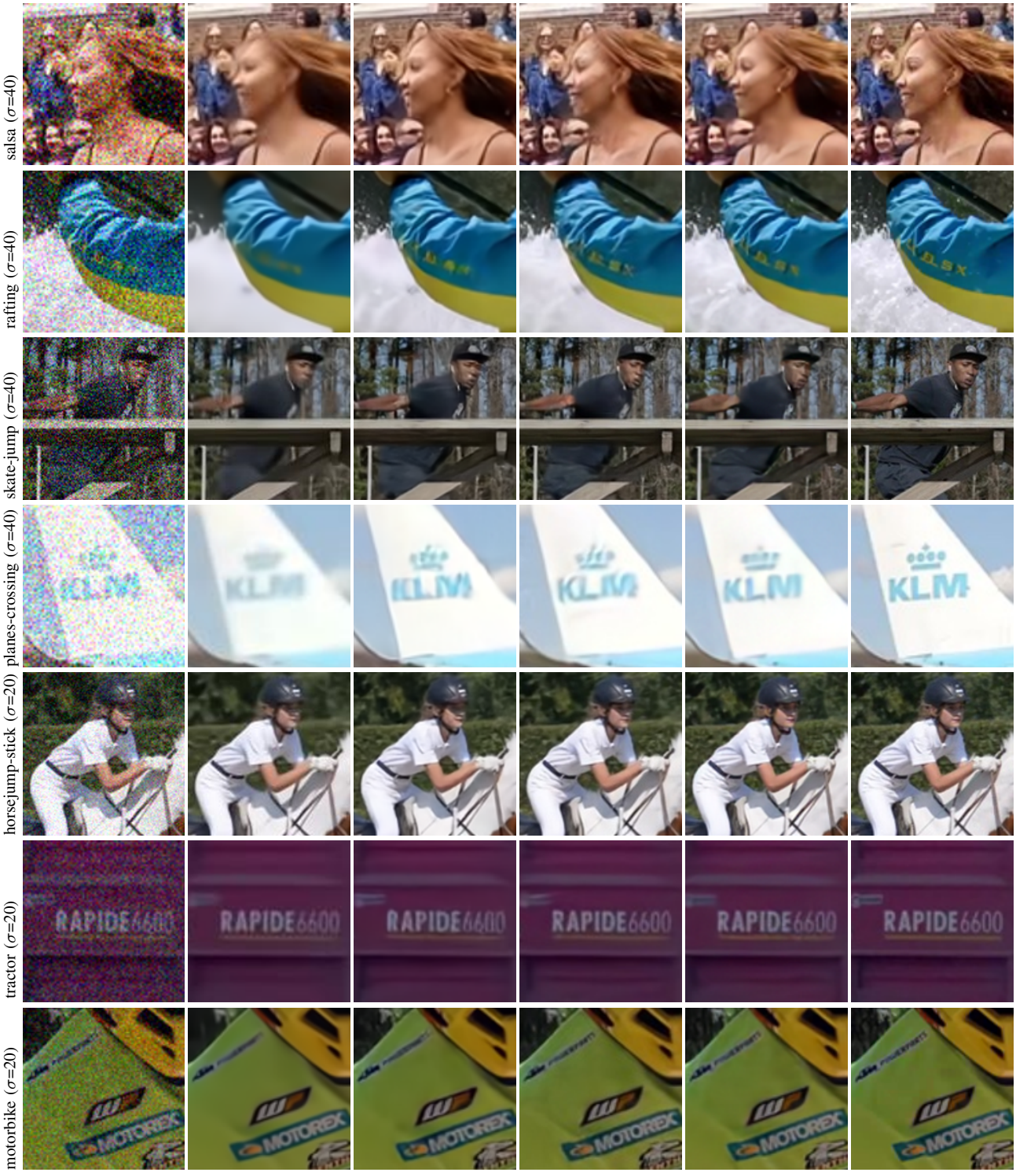

Fig. 4. Visual comparison of denoising results of selected frames.

TABLE III
PSRN(dB) RESULTS OF BASELINES IN ABLATION STUDIES. BEST RESULTS OF EACH COLUMN ARE **BOLDFACED**.

Baseline	σ=10	σ=20	σ=30	σ=40	σ=50	Average
w/o FAB	39.24	36.39	34.59	33.25	32.18	35.13
w/o MRB	39.49	36.61	34.80	33.47	32.40	35.35
w/o noise map	39.52	36.53	34.70	33.35	32.26	35.27
w/o LSTM	39.56	36.63	34.78	33.45	32.33	35.35
Original	**39.88**	**36.82**	**34.96**	**33.58**	**32.47**	**35.52**

following baseline models. (i) w/o FAB: The model is trained without the FAB module. (ii) w/o MRB: The model is trained without the MRB module. (iii) w/o noise map: The model is trained without using noise level maps as the conditional input in the CCLs. (iv) w/o CovnLSTM: The ConvLSTM module in each coupling layer is replaced by a DenseBlock of a similar size. The results of these baselines on the DAVIS test set are listed in Table III. It can be seen that the original IVDNet outperforms all the baselines with noticeable PSNR improvement, which indicates the effectiveness of each key component in our proposed model.

IV. CONCLUSION

In this paper, we proposed an effective DNN for video denoising, which is built upon a progressive decomposition process implemented by conditional coupling layers. The two branches of the constructed coupling layers interact with each other for feature refinement and disentanglement. Meanwhile, they keep all available features for reconstructing the reference frame, due to their invertibility. By incorporating ConvLSTM into the coupling layers, our model can better capture temporal information for disentanglement. In addition, an estimated noise map is used to condition the coupling layers to better handle videos with variant noise levels. In the extensive experiments, our proposed DNN outperformed existing methods both quantitatively and qualitatively.

REFERENCES

[1] Huaian Chen, Yi Jin, Kai Xu, Yuxuan Chen, and Changan Zhu, "Multiframe-to-multiframe network for video denoising," *IEEE Transactions on Multimedia*, vol. 24, pp. 2164–2178, 2021.

[2] Chenyang Qi, Junming Chen, Xin Yang, and Qifeng Chen, "Real-time streaming video denoising with bidirectional buffers," in *Proceedings of ACM International Conference on Multimedia*, 2022, pp. 2758–2766.

[3] 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, 2017.

[4] 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, 2018.

[5] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu, "Residual dense network for image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 7, pp. 2480–2495, 2020.

[6] Yuhui Quan, Mingqin Chen, Tongyao Pang, and Hui Ji, "Self2self with dropout: Learning self-supervised denoising from single image," in *Proceedings of IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1890–1898.

[7] Yang Liu, Zhenyue Qin, Saeed Anwar, Pan Ji, Dongwoo Kim, Sabrina Caldwell, and Tom Gedeon, "Invertible denoising network: A light solution for real noise removal," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13365–13374.

[8] Yuhui Quan, Yixin Chen, Yizhen Shao, Huan Teng, Yong Xu, and Hui Ji, "Image denoising using complex-valued deep cnn," *Pattern Recognition*, vol. 111, pp. 107639, 2021.

[9] 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, 2018.

[10] 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, 2012.

[11] Axel Davy, Thibaud Ehret, Jean-Michel Morel, Pablo Arias, and Gabriele Facciolo, "Non-local video denoising by cnn," *arXiv preprint arXiv:1811.12758*, 2018.

[12] Xinyuan Chen, Li Song, and Xiaokang Yang, "Deep rnns for video denoising," in *Applications of Digital Image Processing XXXIX*. SPIE, 2016, vol. 9971, pp. 573–582.

[13] Matias Tassano, Julie Delon, and Thomas Veit, "Dvdnet: A fast network for deep video denoising," in *IEEE International Conference on Image Processing*. IEEE, 2019, pp. 1805–1809.

[14] Matias Tassano, Julie Delon, and Thomas Veit, "Fastdvdnet: Towards real-time deep video denoising without flow estimation," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1354–1363.

[15] Huan Zheng, Tongyao Pang, and Hui Ji, "Unsupervised deep video denoising with untrained network," in *AAAI Conference on Artificial Intelligence*, 2023.

[16] Tongyao Pang, Huan Zheng, Yuhui Quan, and Hui Ji, "Recorrupted-to-recorrupted: unsupervised deep learning for image denoising," in *Proceedings of IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2043–2052.

[17] Yuhui Quan, Xinran Qin, Tongyao Pang, and Hui Ji, "Dual-domain self-supervised learning and model adaption for deep compressive imaging," in *Proceedings of European Conference Computer Vision*. Springer, 2022, pp. 409–426.

[18] Gregory Vaksman, Michael Elad, and Peyman Milanfar, "Patch craft: Video denoising by deep modeling and patch matching," in *Proceedings of IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2157–2166.

[19] Mingqing Xiao, Shuxin Zheng, Chang Liu, Yaolong Wang, Di He, Guolin Ke, Jiang Bian, Zhouchen Lin, and Tie-Yan Liu, "Invertible image rescaling," in *Proceedings of European Conference on Computer Vision*. Springer, 2020, pp. 126–144.

[20] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[21] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[22] Anurag Ranjan and Michael J Black, "Optical flow estimation using a spatial pyramid network," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4161–4170.

[23] Johannes Ballé, Valero Laparra, and Eero P Simoncelli, "Density modeling of images using a generalized normalization transformation," *arXiv preprint arXiv:1511.06281*, 2015.

[24] Guangyong Chen, Fengyuan Zhu, and Pheng Ann Heng, "An efficient statistical method for image noise level estimation," in *Proceedings of IEEE/CVF International Conference on Computer Vision*, 2015, pp. 477–485.

[25] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[26] Anna Khoreva, Anna Rohrbach, and Bernt Schiele, "Video object segmentation with language referring expressions," in *Proceedings of Asian Conference on Computer Vision*. Springer, 2018, pp. 123–141.

[27] Xavier Glorot and Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[28] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.