TWO-STAGE VIDEO DE-RAINING WITH SPATIO-TEMPORAL FUSION AND ILLUMINATION-INVARIANT DETAIL PRESERVATION

Yufeng Tan, Youjun Xiang, Lei Cai, Pengcheng Wang, Ying Zhang and Yuli Fu*

School of Electronic and Information Engineering, South China University of Technology
Guangzhou 510641, China

ABSTRACT

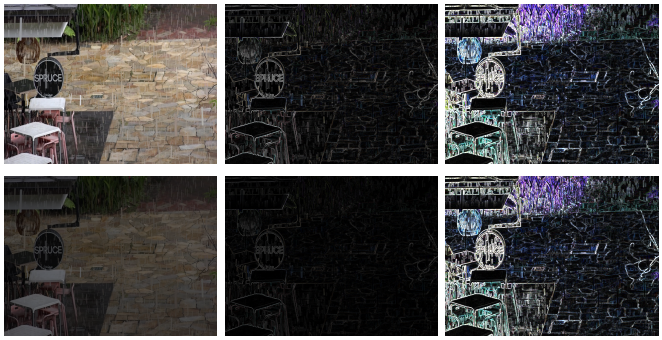
Video de-raining is an important yet highly challenging task in the field of computer vision. Though numerous video de-raining methods are developed with encouraging performance, two major challenges for video de-raining are still unsatisfactorily solved and need to be further investigated as follows: 1) how to sufficiently explore the useful spatio-temporal information from adjacent rainy frames to facilitate the rain removal, and 2) how to well preserve background details even in a video with illumination variance. Regarding the above challenges, this paper specifically develops a new two-stage video de-raining method, which cleverly integrates two typical modules that are beneficial for the video de-raining task, namely *Spatio-Temporal Fusion* (STF) module and *Illumination-Invariant Detail Preservation* (IIDP) module. The STF module is designed to fuse the spatio-temporal information from successive frames effectively, while the IIDP module is developed to deliver the enhanced features from the first stage sub-network to the second stage sub-network to preserve clear edge details of objects. Experimental results demonstrate the superiority of our proposed method over previous state-of-the-arts. The code will be publicly available at https://github.com/mapleTan1113/TSVDN.

Index Terms— Video de-raining, spatio-temporal information, illumination variance, detail preservation

1. INTRODUCTION

Many outdoor vision-based systems such as video surveillance and autonomous vehicle typically require processing and analysis of videos and images captured under severe weather conditions, *e.g.,* rain, snow, haze, and more. These bad weather conditions would undesirably affect the visual quality of images or videos, which further degrades the performance of a vision-based system. Therefore, it is very necessary to develop an effective method to automatically remove these artifacts.

This paper handles the problem of rain removal from a video. In comparison with rain removal from a single image, such as [1–9], video de-raining methods focus on exploring correlation between adjacent frames and further employ their *temporal correlation* to operate the pixel value degraded by rain streaks [4]. However, because of the heavy rain and the fast movement of camera, extracting the spatio-temporal information from successive frames to remove rain streaks remains a formidable challenge. Regarding this challenge, [10] presented SpacCNN for video de-raining, which was an *alignment* method based on super-pixel matching to estimate coherence of video sequence. However, their proposed SpacCNN is very time consuming, because it relies heavily on super-pixel matching, resulting in extracting temporal information inefficiently. Besides, some works [11–14] devote to estimating the *optical flow* in a video to obtain the correlations between adjacent frames. However, the rain streaks in different frames would violate the assumption of brightness constancy constraint to some extent [15], thus these methods could produce invalid optical flow estimation.



(a) (b) (c)

Fig. 1. Two examples of original frames (a) with different illumination conditions, as well as their corresponding edge detail maps produced by the detail extractor from [8] (b) and our Illumination-Invariant Detail Extractor (c).

On the other hand, since the edge details of objects and the rain streaks both represent the high-frequency information and they are overlapped intrinsically in each frame, this may make the edge details of objects being removed together with the rain streaks. To mitigate this issue, [8, 9, 13, 16–18] have proposed their own schemes to remove the rain steaks and meanwhile preserve the edge details of objects. Unfortunately, none of them take into account the effect of illumination changes on the recovery of edge details. During the video capture, the brightness in each frame could be different, owning to the variation of illumination conditions. This could make the detail recovery become more difficult. Considering that *gradient* can characterize the high-frequency edge details of objects, [8] leverage it to explore the edge details of objects in images. However, when the illumination is weakened, the edge details extracted by their method would also be seriously impaired, see the first and second row in both Fig. 1 (a) and Fig. 1 (b) for a comparison. Even though low-light enhancement techniques could help address this issue to some extent, this processing scheme involves running two algorithms one after the other to first enhance the illumination of each video frame and then remove the rain streaks, which is very tedious and time-consuming.

In this work, we propose a two-stage video de-raining network to achieve better de-raining performance and clearer detail preservation. The first-stage sub-network aims to remove the rain streaks

This work is partially supported by Natural Science Foundation of Guangdong Province (2019A1515010861), Guangzhou Technical Project (Grant: 201902020008), and NSFC (Grant: 61471174). **(Corresponding author: Youjun Xiang).*

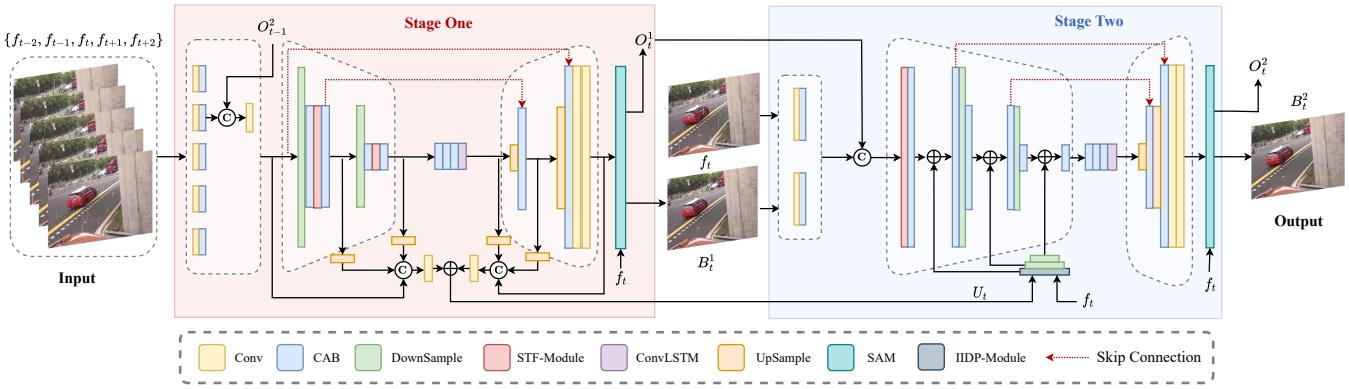


Fig. 2. The overall architecture of our proposed two-stage video de-raining network, where the symbol © denotes the concatenation operation, whereas ⊕ represents the element-wise addition.

in a current frame through fusing contextual feature information extracted from the future and past frames. In order to better extract the spatio-temporal information for video de-raining, we design a Spatio-Temporal Fusion (STF) module and integrate it into our network to adaptively gate and fuse useful spatio-temporal information. The second-stage sub-network devotes to refine the de-rained outcome and restore the missing edge details of objects due to the first-stage de-raining process. To prevent the effect of illumination fluctuations on the recovery of background details, a novel Illumination-Invariant Detail Preservation (IIDP) module is developed to deliver the detail enhanced features from the first-stage sub-network to the second-stage one for preserving more edge details of objects. Experimental results show that our method can achieve clear frame detail preservation and bring the significant performance improvements over the State-Of-The-Art (SOTA) methods.

2. PROPOSED METHOD

The main objective of this work lies in: 1) how to effectively extract the effective spatio-temporal information from successive frames to more sufficiently remove the rain streaks; 2) how to preserve the edge details of objects during the de-raining process, especially for a rainy video captured under the different illuminations. For that, this paper proposes a two-stage video de-raining method which cleverly integrates two typical modules that are favorite to the video de-raining task, namely Spatio-Temporal Fusion (STF) module and Illumination-Invariant Detail Preservation (IIDP) module. The overall architecture of our proposed method is shown in Fig. 2.

2.1. Stage One: De-raining Stage with Spatio-Temporal Fusion

The first stage sub-network is an encoder-decoder structure with skip connection, whose encoder takes five successive frames $\{\boldsymbol{f}_{t-2}, \boldsymbol{f}_{t-1}, \boldsymbol{f}_t, \boldsymbol{f}_{t-1}, \boldsymbol{f}_{t-2}\}$ as input, and make them pass through a series of Convolutional layer, Channel Attention Block (CAB) [19], and DownSample layer to learn high-level contextual features of the input successive frames. Note that, in order to better explore the spatio-temporal information within adjacent frames, we specifically design an effective and lightweight STF module, which is further incorporated into the encoder to adaptively gate and fuse useful spatio-temporal information from these high-level contextual features. And the decoder receives these features characterizing the spatio-temporal information and the high-level contextual features to generate a coarse de-rained frame $\boldsymbol{B}_t^1$, which corresponds to the current frame $\boldsymbol{f}_t$. Besides $\boldsymbol{B}_t^1$, the decoder also delivers an attention augmented feature representation (*i.e.,* $\boldsymbol{O}_t^1$) through a Supervised Attention Module (SAM) [6], which is passed to the second stage sub-network for further processing. Also, the first stage sub-network deliver parts of its contextual features, *i.e.,* $\boldsymbol{U}_t$ extracted from the current frame $\boldsymbol{f}_t$, to the Illumination-Invariant Detail Preservation (IIDP) module in the second stage sub-network, which will be presented in Section 2.2.

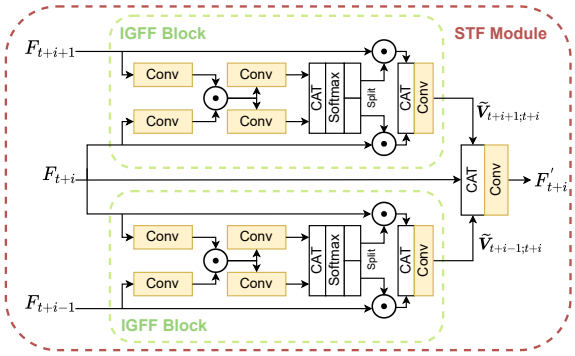


Fig. 3. The structure of Spatio-Temporal Fusion module.

Due to the fast motion of rain raindrops, the rainy region in current frame may be rain-free one in its neighboring frames. With this consideration, STF module fuses the useful spatio-temporal information from future and past frames to help restore clean rain-free frame. The detailed structure of our STF module is depicted in Fig. 3. As shown, the STF module is mainly composed of two parallel Interactive Gating Feature Fusion (IGFF) blocks to respectively fuse the input features $\boldsymbol{F}_{t+i}$ extracted from the frame $\boldsymbol{f}_{t+i}$ and the ones extracted from its adjacent frames $\{\boldsymbol{f}_{t+i-1}, \boldsymbol{f}_{t+i+1}\}$, where $i \in \{-1, 0, 1\}$. Taking the lower IGFF block in Fig. 3 as an example, its computing process can be written as:

$$
\begin{aligned}
&\boldsymbol{V}_{t+i-1;t+i} = \mathrm{Conv}(\boldsymbol{F}_{t+i-1}) \odot \mathrm{Conv}(\boldsymbol{F}_{t+i}),\\
&\{\boldsymbol{V}^1_{t+i-1;t+i}, \boldsymbol{V}^2_{t+i-1;t+i}\} =\\
&\qquad \mathrm{Split}\Big(\mathrm{Softmax}\Big(\mathrm{CAT}\big(\mathrm{Conv}(\boldsymbol{V}_{t+i-1;t+i}); \mathrm{Conv}(\boldsymbol{V}_{t+i-1;t+i})\big)\Big)\Big),\\
&\widetilde{\boldsymbol{V}}_{t+i-1;t+i} =\\
&\qquad \mathrm{Conv}\Big(\mathrm{CAT}\Big(\boldsymbol{F}_{t+i-1} \odot \boldsymbol{V}^1_{t+i-1;t+i}; \boldsymbol{F}_{t+i} \odot \boldsymbol{V}^2_{t+i-1;t+i}\Big)\Big),
\end{aligned}
\tag{1}
$$

where $\odot$ denotes an element-wise multiplication, $\mathrm{Conv}(\cdot)$ denotes a convolution operation, whereas $\mathrm{CAT}(;)$, $\mathrm{Softmax}(\cdot)$, and $\mathrm{Split}(\cdot)$ respectively denote the Concatenation, Softmax function, and Feature-group Split, and these three operations are all conducted along the channel axis. Clearly, according to Eq.(1), the output of our IGFF block (*i.e.,* $\widetilde{\boldsymbol{V}}_{t+i-1;t+i}$) should be able to integrate the

spatio-temporal information from $\boldsymbol{f}_{t+i-1}$ and $\boldsymbol{f}_{t+i}$.

Likewise, the upper IGFF module in Fig. 3 concludes the same processing as the lower one. Thus, its output (*i.e.*, $\widetilde{\boldsymbol{V}}_{t+i+1;t+i}$) is supposed to fuse the spatio-temporal information from $\boldsymbol{f}_{t+i}$ and $\boldsymbol{f}_{t+i+1}$. Finally, the proposed STF module fuses the input features $\boldsymbol{F}_{t+i}$ with the spatio-temporal features, *i.e.,* $\widetilde{\boldsymbol{V}}_{t+i-1;t+i}$ and $\widetilde{\boldsymbol{V}}_{t+i+1;t+i}$ delivered by the lower and upper IGFF blocks, to get the fused spatio-temporal features, as follows:

$$\boldsymbol{F}^{'}_{t+i} = \text{Conv}(\text{CAT}(\widetilde{\boldsymbol{V}}_{t+i-1;t+i}; \boldsymbol{F}_{t+i}; \widetilde{\boldsymbol{V}}_{t+i+1;t+i})). \tag{2}$$

Due to $i \in \{-1, 0, 1\}$, thus after the first STF module in the first stage, there will produce three types of fused spatio-temporal features, *i.e.,* $\boldsymbol{F}^{'}_{t-1}$, $\boldsymbol{F}^{'}_{t}$ and $\boldsymbol{F}^{'}_{t+1}$, corresponding to the current frame $\boldsymbol{f}_t$ and its adjacent frames $\boldsymbol{f}_{t-1}$ and $\boldsymbol{f}_{t+1}$. Similarly, after the second STF module, there will produce further fused spatio-temporal features $\boldsymbol{F}^{''}_{t}$, which corresponds to the current frame $\boldsymbol{f}_t$.

2.2. Stage Two: Refinement Stage with Illumination-Invariant Detail Preservation

During the first stage de-raining process, the edge details of objects could be removed together with the rain streaks, this may result in the loss of the background details. Considering this, the second stage sub-network takes the current frame $\boldsymbol{f}_t$ and the coarse de-rained frame $\boldsymbol{B}^1_t$ as input, and pass them through a Convolutional layer and a Channel Attention Block (CAB) [19]. After that, their output features are concatenated with the attention augmented feature $\boldsymbol{O}^1_t$ to integrate more contextual features for the de-raining refinement and the detail restoration, as shown in Fig. 2.

On the other hand, the light fluctuation, such as Illumination changes within the captured video, may make it difficult to preserve the edge details of objects, as shown in Fig. 1 (b). Thus, in order to prevent the impact of illumination variation on the edge detail recovery, we specifically design an Illumination-Invariant Detail Preservation (IIDP) module to selectively filter the illumination components. The detailed structure of the proposed IIDP module is shown in Fig. 4, and its computing process can be expressed as:

$$\begin{aligned} \boldsymbol{R}_t &= \text{Conv}(\boldsymbol{U}_t) \oplus \boldsymbol{f}_t, \\ \boldsymbol{M}_t &= \text{IIDE}(\boldsymbol{R}_t), \\ \widetilde{\boldsymbol{U}}_t &= \text{IGFF}(\boldsymbol{U}_t; \text{CAB}(\text{Conv}(\boldsymbol{M}_t))), \end{aligned} \tag{3}$$

where $\boldsymbol{U}_t$ denotes the contextual features of current frame $\boldsymbol{f}_t$, which are derived by the first stage sub-network, whereas IIDE$(\cdot)$ represents the process of illumination-invariant detail extractor, whose formulation is defined as:

$$\text{IIDE}(\boldsymbol{R}_t) = \arctan\left(\sqrt{\left(\frac{\boldsymbol{R}_t * \boldsymbol{P}^x_t}{\max(\boldsymbol{R}_t, \epsilon)}\right)^2 + \left(\frac{\boldsymbol{R}_t * \boldsymbol{P}^y_t}{\max(\boldsymbol{R}_t, \epsilon)}\right)^2}\right), \tag{4}$$

where $\boldsymbol{P}^x_t = \boldsymbol{R}_t(x+1, y) - \boldsymbol{R}_t(x-1, y)$ and $\boldsymbol{P}^y_t = \boldsymbol{R}_t(x, y+1) - \boldsymbol{R}_t(x, y-1)$, which respectively denote the horizontal and vertical gradient at pixel position (x, y). $*$ is a convolution operation, and ϵ is a threshold which helps guarantee the numerical stability. To demonstrate the effect of illumination-invariant detail extractor, Fig. 1 (c) gives two examples of edge detail maps derived by our proposed IIDP module. As can be seen, even in different illumination conditions, the proposed IIDP module can still well preserve the edge details of objects. This is due to the fact that the illumination components can be effectively filtered out through the predefined convolution masks (*i.e.,* $\boldsymbol{P}^x_t$ and $\boldsymbol{P}^y_t$) and the internal normalization operation in Eq.(4).

After the IIDP module, the output features $\widetilde{\boldsymbol{U}}_t$ are down-sampled to different scales and then embedded into the encoder of our second stage sub-network via the element-wise addition operations. This can effectively enhance the contextual features to preserve more edge details. Hence, the decoder can decode these detail-enhanced contextual features to produce the final de-rained frame which contains more background details. Note that, similar to the first stage sub-network, we also exploit a SAM [6] to output the final de-rained frame and an attention augmented feature representation $\boldsymbol{O}^2_t$ used for next time step.

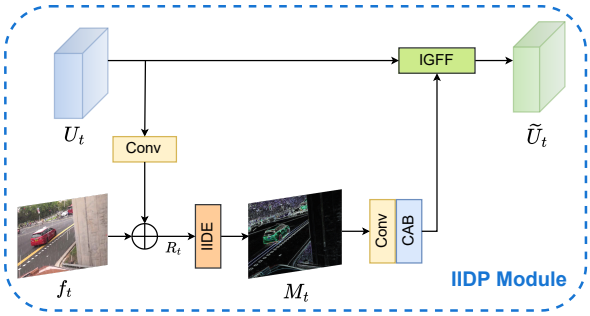


Fig. 4. The structure of Illumination-Invariant Detail Preservation module.

2.3. Loss Functions

In order to achieve the best de-raining performance and direct the network to preserve more edge details of objects, we jointly use multiple loss functions, including Charbonnier loss [20], SSIM loss and edge loss to train our two stages video de-raining network, as follows:

$$\mathcal{L}_{\text{total}} = \sum_{s=1}^{2}\left[\mathcal{L}_{\text{char}}\left(\mathbf{B}^s_t, \mathbf{G}_t\right) + \mathcal{L}_{\text{ssim}}\left(\mathbf{B}^s_t, \mathbf{G}_t\right)\right] + \lambda \mathcal{L}_{\text{detail}}\left(\mathbf{R}_t, \mathbf{G}_t\right), \tag{5}$$

where s denotes the number of stages, $\boldsymbol{G}_t$ represents the Ground Truth of rain-free frame, and

$$\mathcal{L}_{\text{char}}(\mathbf{B}^s_t, \mathbf{G}_t) = \sqrt{\|(\mathbf{B}^s_t) - (\mathbf{G}_t)\|^2 + \varepsilon^2}, \tag{6}$$

$$\mathcal{L}_{\text{ssim}}(\mathbf{B}^s_t, \mathbf{G}_t) = 1 - \text{SSIM}(\mathbf{B}^s_t, \mathbf{G}_t), \tag{7}$$

$$\mathcal{L}_{\text{detail}}(\mathbf{R}_t, \mathbf{G}_t) = \sqrt{\|\text{IIDE}\,(\mathbf{R}_t) - \text{IIDE}(\mathbf{G}_t)\|^2 + \varepsilon^2}, \tag{8}$$

3. EXPERIMENTS

In this section, we conducted video de-raining experiments on three widely-used video datasets, including NTURain [10], RainSynLight25, and RainSynComplex25 [16]. We compare the proposed method with multiple recently-developed methods, including FastDerain [21], J4RNet [16], SpacCNN [10], FCRNet [13], and ESTINet [17]. For the performance evaluation, we used two commonly-used metrics including Peak Signal to Noise Ration (PSNR) and Structure Similarity Index (SSIM) [22].

Implementation details: We trained our de-raining model using the Pytorch framework on two NVIDIA GTX 1080Ti GPUs with a mini-batch size of 4 for 200 epochs. The initial learning rate was set to 2×10^{-3}, and was gradually decreased to 1×10^{-6} according to the cosine annealing strategy [23]. Besides, the hyperparameters, including ϵ in Eq.(4), λ in Eq.(5), ε in Eq.(6) and Eq.(8) were set to 10^{-6}, 0.5, 10^{-3}, and 10^{-3}, respectively.

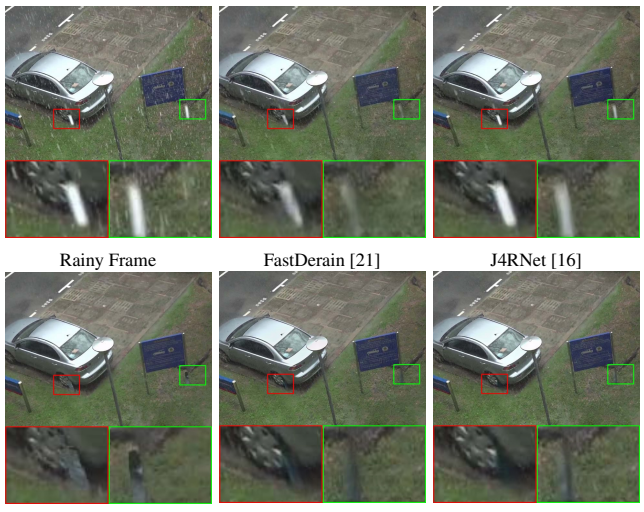
Comparison with SOTA Methods: Table 1 shows the quantitative results between our method and several SOTA video deraining methods. As can been seen, our method can deliver better de-raining performance on all three datasets and consistently outperform these competing methods. In addition, Fig. 5 and Fig. 6 display the visual de-raining results by different methods on a synthetic and a real-world video frame. We can observe that those competing methods

(a) Rainy Frame (b) FastDerain [21] (c) J4RNet [16] (d) SpacCNN [10] (e) ESTINet [17] (f) **Ours** (g) Ground Truth

Fig. 5. Visual comparison on synthetic rainy frame selected from RainSynComplex25 [16].

Table 1. Quantitative comparison on synthesized datasets.

Datasets	Metric	FastDerain	J4RNet	SpacCNN	FCRNet	ESTINet	**Ours**
RainSynLight25	PSNR	29.39	32.96	32.29	35.80	36.32	**37.24**
	SSIM	0.8665	0.9434	0.9137	0.9622	0.9594	**0.9763**
RainSynComplex25	PSNR	19.25	24.13	21.33	27.72	28.57	**32.88**
	SSIM	0.5386	0.7163	0.5889	0.8239	0.8261	**0.9359**
NTURain	PSNR	30.53	32.14	33.11	36.05	37.61	**39.10**
	SSIM	0.9263	0.9480	0.9474	0.9676	0.9712	**0.9772**



Rainy Frame FastDerain [21] J4RNet [16]

SpacCNN [10] ESTINet [17] **Ours**

Fig. 6. Visual comparison on a real rainy frame from NTURain [10].

either fail to effectively remove the rain streaks or lose some background details, while our method can more clearly remove the rain streaks and also better preserve the details of objects.

Ablation Study: Besides the qualitative and quantitative comparisons, we also perform an ablation study on the RainSynComplex25 [16] dataset to validate the proposed Spatio-Temporal Fusion (STF), Illumination-Invariant Detail Preservation (IIDP) modules and two-stage architecture. The corresponding results are shown in Table 2, where the 3DCNN denotes the baseline that adopts a 3D convolution to replace our STF module. As can be seen, the de-raining performance decreases remarkably after removing our specifically-designed STF and IIDP modules. By adding our STF module to replace the 3D convolution, the performance is significantly improved, as can be seen from Fig. 7 that our STF module can more completely remove rain streaks. The comparison between M_2 and M_3 shows that adding Stage Two can further improve the performance, and using the IIDP module (*i.e.*,M_4) achieves the best performance. Furthermore, Fig. 8 shows two rainy frames with different illuminations and their corresponding de-rained results produced by our de-raining network with (w) and without (w/o) IIDP module. One can observe that, even when the illumination is changed, the proposed method with IIDP module can still well preserve the background details after de-raining.

Table 2. Comparison of PSNR and SSIM results by different variants of our method on RainSynComplex25 [16] dataset.

	3DCNN	STF	Stage Two	IIDP	PSNR	SSIM
M_1	✓	✗	✗	✗	31.21	0.9174
M_2	✗	✓	✗	✗	31.63	0.9208
M_3	✗	✓	✓	✗	32.52	0.9332
M_4	✗	✓	✓	✓	32.88	0.9359

Rainy Frame M_1 M_2 M_3

Fig. 7. Visual comparison on a rainy frame from RainSynComplex25 [16] by combining different components.

Rainy Frame w/o IIDP w IIDP

Fig. 8. Two rainy frames with different illuminations and their derained results derived by our method with and without IIDP module.

Table 3. Average running time of different methods (in sec) to remove rain in a video with the resolution of 832×512 per frame.

Methods	FastDerain	J4RNet	SpacCNN	FCRNet	ESTINet	**Ours**
Time	0.3861	0.8401	4.1698	0.8974	0.3714	**0.1253**

Comparison of Running Time: We further compare our model with other de-raining methods on running speed for processing a single video frame on a test video with the resolution of 832×512. The average running time of different methods are shown in Table 3. For a fair comparison, all of compared methods were run in the GPU mode with the same environment. Obviously, our method achieves faster processing speed than those methods compared against.

4. CONCLUSION

In this paper, we proposed a two-stage network with Spatio-Temporal Fusion (STF) and Illumination-Invariant Detail Preservation (IIDP) modules for video de-raining. In brief, we specifically designed STF and IIDP modules that are beneficial for the video de-raining task. STF module is designed to extract reliable spatio-temporal information from adjacent frames, and IIDP module allows the network to better restore the details of objects under different illumination conditions after de-raining. Both quantitative and qualitative results demonstrate the superior performance of our method.

5. REFERENCES

[1] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng, "Progressive image deraining networks: A better and simpler baseline," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3937–3946.

[2] Shen Zheng, Changjie Lu, Yuxiong Wu, and Gaurav Gupta, "Sapnet: Segmentation-aware progressive network for perceptual contrastive deraining," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 52–62.

[3] Yizhou Li, Yusuke Monno, and Masatoshi Okutomi, "Single image deraining network with rain embedding consistency and layered lstm," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 4060–4069.

[4] Lei Cai, Yuli Fu, Tao Zhu, Youjun Xiang, Ying Zhang, and Huanqiang Zeng, "Joint depth and density guided single image de-raining," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 7, pp. 4108–4121, 2021.

[5] Lei Cai, Yuli Fu, Wanliang Huo, Youjun Xiang, Tao Zhu, Ying Zhang, Huanqiang Zeng, and Delu Zeng, "Multi-scale attentive image de-raining networks via neural architecture search," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[6] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao, "Multi-stage progressive image restoration," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14821–14831.

[7] Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen, "Hinet: Half instance normalization network for image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 182–192.

[8] Ying Zhang, Youjun Xiang, Lei Cai, Yuli Fu, Wanliang Huo, and Junjun Xia, "Single image de-raining with high-low frequency guidance," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 2330–2334.

[9] Qiaosi Yi, Juncheng Li, Qinyan Dai, Faming Fang, Guixu Zhang, and Tieyong Zeng, "Structure-preserving deraining with residue channel prior guidance," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4238–4247.

[10] Jie Chen, Cheen-Hau Tan, Junhui Hou, Lap-Pui Chau, and He Li, "Robust video content alignment and compensation for rain removal in a cnn framework," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6286–6295.

[11] Jin-Hwan Kim, Jae-Young Sim, and Chang-Su Kim, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2658–2670, 2015.

[12] Jiaying Liu, Wenhan Yang, Shuai Yang, and Zongming Guo, "D3r-net: Dynamic routing residue recurrent network for video rain removal," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 699–712, 2018.

[13] Wenhan Yang, Jiaying Liu, and Jiashi Feng, "Frame-consistent recurrent video deraining with dual-level flow," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1661–1670.

[14] Wenhan Yang, Robby T Tan, Shiqi Wang, and Jiaying Liu, "Self-learning video rain streak removal: When cyclic consistency meets temporal correspondence," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1720–1729.

[15] Ruoteng Li, Robby T Tan, Loong-Fah Cheong, Angelica I Aviles-Rivero, Qingnan Fan, and Carola-Bibiane Schonlieb, "Rainflow: Optical flow under rain streaks and rain veiling effect," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7304–7313.

[16] Jiaying Liu, Wenhan Yang, Shuai Yang, and Zongming Guo, "Erase or fill? deep joint recurrent rain removal and reconstruction in videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3233–3242.

[17] Kaihao Zhang, Dongxu Li, Wenhan Luo, Wenqi Ren, and Wei Liu, "Enhanced spatio-temporal interaction learning for video deraining: A faster and better framework," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[18] Xinwei Xue, Ying Ding, Long Ma, Yi Wang, Risheng Liu, and Xin Fan, "Temporal rain decomposition with spatial structure guidance for video deraining," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 2015–2019.

[19] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 286–301.

[20] Pierre Charbonnier, Laure Blanc-Feraud, Gilles Aubert, and Michel Barlaud, "Two deterministic half-quadratic regularization algorithms for computed imaging," in *Proceedings of 1st International Conference on Image Processing*. IEEE, 1994, vol. 2, pp. 168–172.

[21] Tai-Xiang Jiang, Ting-Zhu Huang, Xi-Le Zhao, Liang-Jian Deng, and Yao Wang, "Fastderain: A novel video rain streak removal method using directional gradient priors," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 2089–2102, 2018.

[22] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[23] Ilya Loshchilov and Frank Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.