Enhanced Real-Time Intermediate Flow Estimation for Video Frame Interpolation

Minseop Kim and Haechul Choi*

Department of Multimedia Engineering, Hanbat National University, Daejeon, Republic of Korea
E-mail: choihc@hanbat.ac.kr
**Corresponding Author*

Received 15 April 2021; Accepted 27 July 2021;
Publication 06 November 2021

Abstract

Recently, the demand for high-quality video content has rapidly been increasing, led by the development of network technology and the growth in video streaming platforms. In particular, displays with a high refresh rate, such as 120 Hz, have become popular. However, the visual quality is only enhanced if the video stream is produced at the same high frame rate. For the high quality, conventional videos with a low frame rate should be converted into a high frame rate in real time. This paper introduces a bidirectional intermediate flow estimation method for real-time video frame interpolation. A bidirectional intermediate optical flow is directly estimated to predict an accurate intermediate frame. For real-time processing, multiple frames are interpolated with a single intermediate optical flow and parts of the network are implemented in 16-bit floating-point precision. Perceptual loss is also applied to improve the cognitive performance of the interpolated frames. The experimental results showed a high prediction accuracy of 35.54 dB on the Vimeo90K triplet benchmark dataset. The interpolation speed of 84 fps was achieved for 480p resolution.

Keywords: Video frame interpolation, optical flow estimation, contextual information, multiscale fusion.

Journal of Web Engineering, Vol. 20_8, 2413–2432.
doi: 10.13052/jwe1540-9589.2089

1 Introduction

As the influence of video streaming platforms has recently expanded, the video streaming market is growing rapidly worldwide. This rapid growth of the video streaming market has contributed to the demand for high-quality video content. Demand for videos with a high frame rate has been increasing significantly due to the spread of displays with a high refresh rate. Despite this demand, as videos with a high frame rate can only be recorded using expensive equipment, research on converting the existing videos with a low refresh rate into a high refresh rate is actively underway.

Video frame interpolation (VFI) is a traditional temporal quality enhancement technique that generates new frames between two frames using consecutive frames, a representative technique in slow-motion video generation. Recently, as deep learning-based temporal quality improvement technology has been researched, various benchmark datasets, such as Vimeo90K, Middlebury, and Adobe240fps, have been released in the VFI field, and various algorithms have been proposed.

Sepconv [1] and Adaconv [2] are representative optical flow-based algorithms, which predict convolutional kernels to interpolate intermediate frames. Guaranteeing real-time processing is a challenge because these methods are designed based on the complex U-Net structure to calculate the intermediate optical flow and transmit the contextual data of the previous convolutional layer. Real-time intermediate flow estimation (RIFE) for VFI [3] has reduced computational complexity significantly by applying a monodirectional intermediate optical flow prediction network. However, the monodirectional intermediate optical flow prediction may have a disadvantage in that the prediction accuracy and inference performance are degraded when interpolating more than one intermediate frame between two consecutive frames.

This paper proposes a fast VFI method that predicts bidirectional intermediate optical flow for accurate interpolation and includes strategies for real time processing. The proposed method introduces three parts as follows. First, it expands the monodirectional intermediate optical flow prediction of RIFE to bidirectional prediction for high interpolation frame quality. The intermediate optical flow estimation network is learned in both directions, so it is possible to learn more elaborately than in one direction. Second, it proposes a one-shot structure and a half-precision implementation for fast inference. The runtime is significantly shortened by predicting multiple frames with one inference and implementing parts of network in a

half-precision. Finally, it applies perceptual loss using the output of the pretrained network feature map for the cognitive prediction accuracy.

2 Related Work

The VFI method has been extensively studied. In this section, the previously proposed research on VFI are described. Early VFI algorithms used the traditional block-level prediction method [4] due to operational limitations. This method has limitations because it predicts the motion vector by dividing the image into blocks.

In the early days when deep learning technology was proposed, kernel-based algorithms were proposed instead of optical flow-based algorithms [2, 5–8]. The motion of the object is not accurately predicted because the occlusion is not considered. This method is relatively inaccurate because the network does not predict the motion of an object and cannot handle the occlusion of the background.

A method of predicting the optical flow and interpolated frames at the pixel level is studied to improve accuracy. An additional structure is proposed to directly predict the optical flow and handle occlusion using a U-Net-shaped [9] network [5, 6, 10, 12]. This method can check the learning progress of the optical flow [3]. Because the context information of the previous layer is inherited, an optical flow with relatively high accuracy can be obtained. However, the complexity is increased because numerous convolutional layers must be stacked. As a result, it is difficult to achieve real-time interpolation.

Quadratic Video Interpolaton (QVI) and Enhanced Quadratic Video Interpolation (EQVI) [13, 14] predicted the optical flow using a curve assuming that the object does not move in a straight line. However, as the network inputs four frames to predict the optical flow of curved objects, the network cannot cope with using only two frames as input.

In addition, a simple and efficient method for predicting optical flow from the U-Net structure has been proposed [3, 15]. The author of SoftSplat [15] proposed a frame interpolation method using forward warping instead of the existing backward warping. The performance does not decrease even if multiple frames are interpolated simultaneously using the feature map of the pyramid structure. Huang et al. proposed a method in that a simple optical flow estimation network is directly learned with a semi-supervised learning. It has a residual block [16, 17] and a coarse-to-fine structure. As a result,

the inference network speed is increased, and the interpolation quality is also improved.

Recently, a method of predicting the final intermediate frame directly using the channel attention structure and Pixel-shuffle [2] without using the optical flow directly was proposed [18]. Choi et al. demonstrated that the attention structure can replace the optical flow network without using the optical flow directly, and the interpolation performance is also sufficient compared to other methods using the optical flow.

In addition, dataset acquisition method based on cycleGAN was proposed to obtain more data in limited training datasets [19].

3 Proposed Architecture

The RIFE algorithm is assessed in Section 3.1. The enhanced bidirectional intermediate optical flow to complement RIFE is proposed in Section 3.2. The propose method to reduce the processing speed is introduced in Section 3.3. Finally, a strategy for learning the proposed network is addressed.

3.1 Reassessing Real-Time Intermediate Flow Estimation for Video Frame Interpolation

(1) Indirect intermediate optical flow estimation

The main purpose of VFI is to generate intermediate frames at the position of the time $t \in [0, 1]$ between two consecutive frames I_0 and I_1 as input. The traditional method is to find the bidirectional optical flow $F_{0\to1}, F_{1\to0}$ between I_0 and I_1 and then linearly combine them to produce an intermediate optical flow $F_{\mathrm{t}\to0}, F_{t\to1}$. Finally, the intermediate optical flow and original videos I_0 and I_1 are used as input to the backward warping function to generate the final intermediate frame I_t. This method has deteriorating image quality by generating artifacts at the motion boundary when it generates an inaccurate intermediate optical flow. It is difficult to apply to real-time applications because it passes through many convolutional layers. A network, such as U-Net [9], which is efficient for capturing the context of input frames, was used to create a bidirectional optical flow [10], but this is also difficult to process in real time in a general environment.

(2) Direct intermediate optical flow estimation

The existing methods can predict bidirectional optical flow and obtain intermediate optical flow. However, RIFE directly predicts the intermediate flow

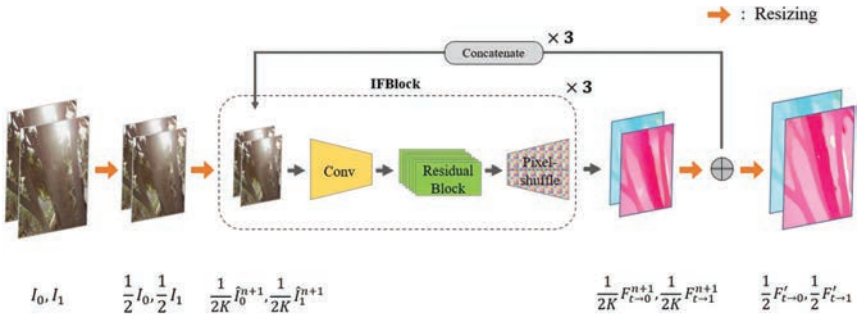

Figure 1 IFNet structure of RIFE, where $K(K \in (4, 2, 1))$ is the resizing variable for the input frame and intermediate optical flow, I_0 and I_1 are converted from low to high resolution according to K through a bilinear interpolation method, n denotes index of resized resolution layer according to K, $\hat{I}^n$ represents the low-resolution image to be input to the intermediate optical flow network, F^n is the coarse optical flow of the n-th resized resolution layer, and F' denotes the intermediate optical flow for final interpolation frame.

to reduce artifacts on the motion boundary and reduce computational complexity. We proposed a coarse-to-fine network that considers the context of the input frame using the residual block [16]:

$$F_{t\to0} = f(I_0,\ I_1), \tag{1}$$

where f denotes the coarse-to-fine optical flow estimation network using the residual block. For a simple structure, the intermediate optical flow in the opposite direction was calculated as follows:

$$F_{t\to1} = -\frac{1-t}{t}F_{t\to0}. \tag{2}$$

In addition, the interpolation performance was improved by directly using the optical flow ground truth $F_{t\to0}^{GT}$ and, $F_{t\to1}^{GT}$ predicted by the pretrained optical flow network LiteFlowNet [11, 20]. However, because most objects or backgrounds can have nonlinear motion, the prediction quality deteriorates as the temporal interval between I_0, I_1 and I_t increases. When predicting multiple frames within time $t \in [0, 1]$, the computational complexity may increase because RIFE has a recursive structure to be described in Section 3.3. Therefore, we reduced this problem using the RIFE structure.

3.2 Enhanced Bidirectional Intermediate Optical Flow Prediction

The proposed method is based on Intermediate optical flow estimation network (IFNet), an intermediate optical flow network of RIFE, and improves

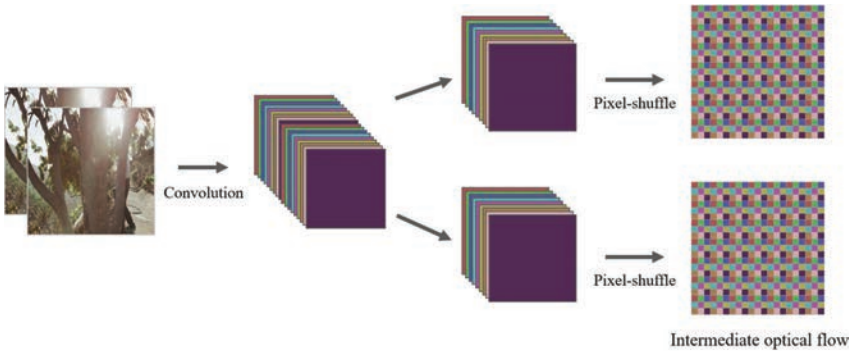

Figure 2 Details of the optical flow output of IFNet.

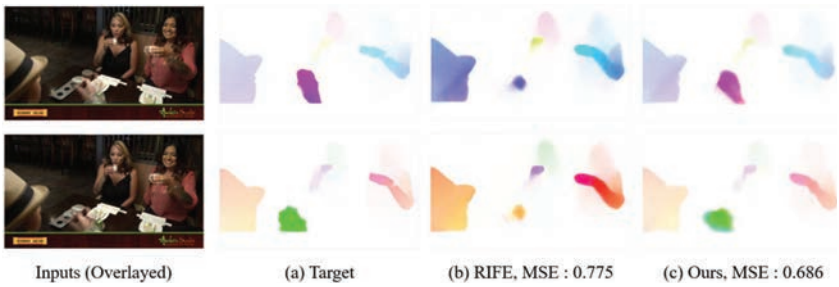

Figure 3 Visualized optical flow maps.

the interpolation accuracy through bidirectional intermediate optical flow estimation. Figure 1 describes the network structure for intermediate optical flow estimation. A residual block was used instead of the U-Net structure for maintaining the context information of the previous layer, which was primarily used for VFI optical flow estimation. In addition, by adopting a coarse-to-fine structure to predict the intermediate optical flow, the optical flow gradually improves to a higher resolution. The output of the last convolutional layer was applied to $F_{t\to 0}$ and $F_{t\to 1}$ using the Pixel-shuffle method. For fast optical flow estimation, instead of using two networks, the layer output was doubled compared to RIFE as shown in Figure 2. This structure has the advantage of more accurately predicting the intermediate optical flow while minimizing the computational increase and providing more precise supervised learning for the intermediate optical flow.

Figure 3 shows the visual comparison between the optical flow prediction result of IFNet and RAFT [21], which exhibits the most improved optical flow estimation score. As observed in Figure 3, the intermediate optical

Table 1 Comparison on 16-bit floating-point precision

Method	LiteFlowNet	RAFT*	RIFE	Ours
Runtime	2×137 ms	2×122 ms	1×122 ms	2×72 ms

*Iterations are set to 20, using the normal size model.

flow result of the proposed method trained with Vimeo90K triplet dataset is learned closer to RAFT than IFNet with RIFE. From the results, bidirectional estimation is more suitable for expressing the flow of a curved object than monodirectional estimation.

In Table 1, the results of the optical flow runtime of networks for one interpolation are compared. The experiment was performed on Nvidia RTX 2080 TI using a 720p resolution image. The comparison results still have a significantly shorter inference speed than other optical flow estimation networks, and no significant difference in speed exists compared to IFNet with RIFE.

3.3 Strategies to Improve the Intermediate Frame Inference Time

(1) Sixteen-bit floating-point operation for frame interpolation

Half-precision is an implementation technique to increase the inference speed using the half-precision floating point (floating point 16 bit, FP16) instead of a single floating-point 32 bit (FP32). In general, the FP16 precision is used for mixed-precision training, which is a method to improve learning speed. It can reduce interpolation speed and memory usage at a cost of inference quality. Table 2 lists the results of FP32 precision and FP16 precision in the Vimeo90K dataset. The Nvidia RTX series graphics card maximizes speed improvement in FP16. Moreover, FP16 is the result of forcibly converting to FP16 from the FP32 network without mixed-precision training. Although the interpolation accuracy decreased, the inference speed is greatly improved. From the above results, using FP16 precision in a limited environment, we can access more real-time inference speed through the maximized inference speed. Additionally, the result of training the FP16 network using mixed-precision training indicates that more improved results can be expected with relatively few resources.

(2) Multiple frame interpolation architecture without recursive structure

RIFE has the outline of network architecture like Figure 4. FusionNet includes a context extractor network to which an intermediate optical flow obtained by IFNet and warped frames are fed. The proposed method is based

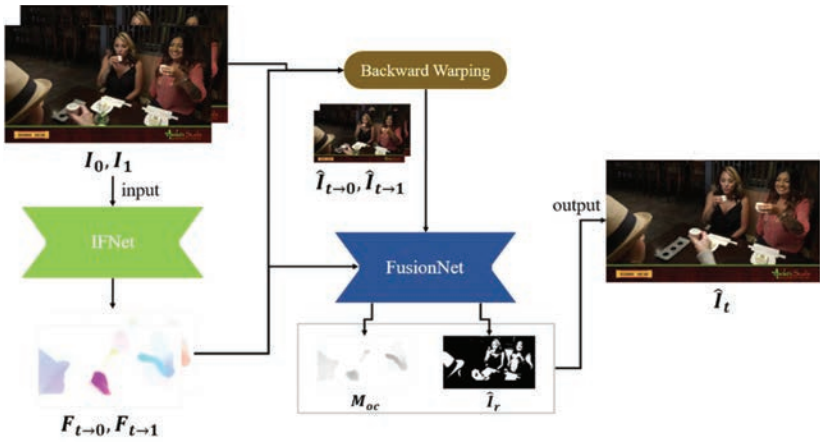

Figure 4 Outline of network architecture. M_{oc} is occlusion map to fuse warped frames, $\hat{I}_r$ denotes residual frame for the output frame refinement.

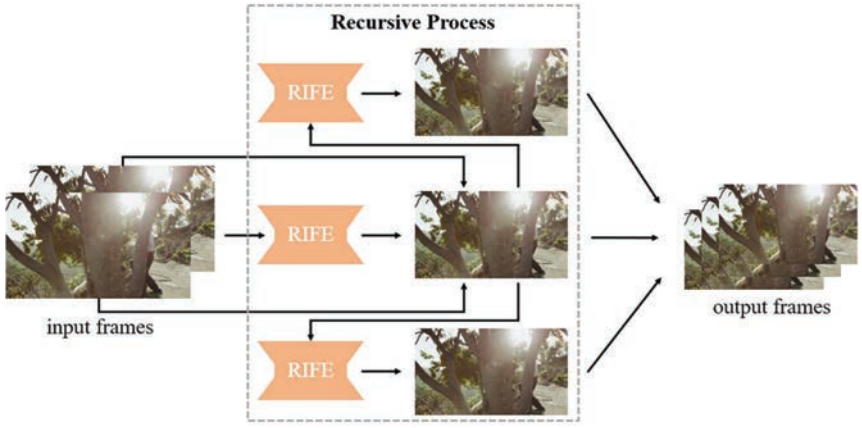

Figure 5 Multiple interpolations in the RIFE structure.

on the network architecture of RIFE. RIFE reuses the previously interpolated frame and needs to produce each intermediate optical flow for multiple frame interpolation using IFNet, which is noted as recursive structure, as illustrated in Figure 5. For example, after interpolating the frame with t equal to 1/2, the frames with t equal to 1/4 and 3/4 respectively are estimated using the interpolated frame with t equal to 1/2. RIFE performs IFNet and FusionNet multiple times, and it also has a delay waiting for interpolating the middle frame. These works require a substantial execution time.

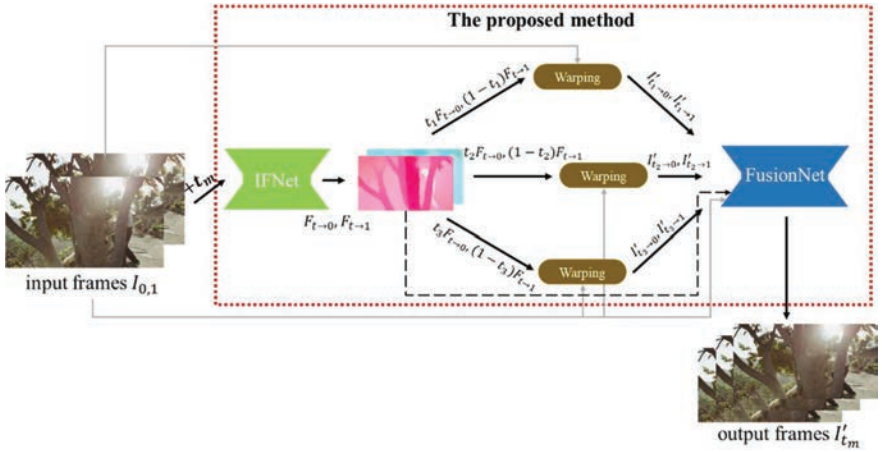

Figure 6 Flowchart of the proposed method with one-shot structure.

To solve this problem, the propose method predicts an intermediate optical flow using IFNet only once and obtains each optical flow for the multiple frame interpolation by a simple calculation with the intermediate optical flow, which is noted as one-shot structure as shown in Figure 6.

$$[F_{t\to 0},\ F_{t\to 1}] = N_F(I_0,\ I_1), \tag{3}$$

where N_F denotes intermediate optical flow estimation network like IFNet. With I_0 and I_1, the intermediate optical flow estimation network obtains $F_{t\to 0}$ and $F_{t\to 1}$ simultaneously. These $F_{t\to 0}$ and $F_{t\to 1}$ result in the bidirectional intermediate optical flow. When interpolating M frames, $t_m \in [0, 1]$ denotes time of the m-th frame to be interpolated. Using the bidirectional intermediate optical flow and t_m, warped frames $I_{t_m\to 0}$ and $I_{t_m\to 1}$ are generated. Note that the same bidirectional intermediate optical flow is used for interpolating multiple frames, which solves the problem described above. With the warped frames and the bidirectional intermediate optical flow, FusionNet makes occlusion map and residual frame. The occlusion map represents occlusion of objects and the residual frame is used for the output frame refinement. Finally, interpolated frame is achieved with the occlusion map and the residual frame.

3.4 Proposed Training Loss Function

Given consecutive input frames I_0 and I_1 and the intermediate frame between consecutive frames, where $t = 1/2$, the proposed overall loss function

consists of four loss functions:

$$L = \lambda_r L_r + \lambda_f L_f + \lambda_c L_c + \lambda_p L_p. \tag{4}$$

(1) Reconstruction loss

The reconstruction loss, L_r, evaluates the reconstruction quality of the interpolated frame:

$$L_r = \|I_t - I_t^{GT}\|_1. \tag{5}$$

L_r is a loss function commonly used when evaluating a VFI network, and the final interpolated frame is compared with the original pixel by pixel. In this paper, λ_f was set to 1.

(2) Optical flow loss

The optical flow loss, L_f, evaluates the intermediate optical flow:

$$L_f = \|F_{t\to 0}^T - F_{t\to 0}^S\|_1 - \|F_{t\to 1}^T - F_{t\to 1}^S\|_1, \tag{6}$$

where F^T is an optical flow map from a well-trained optical flow estimation network. RAFT [2] is used as a well-trained optical flow network. F^S is the estimated intermediate optical flow map used for frame interpolation. L_f is a way of knowledge distillation. If a relatively tiny intermediate optical flow network is learned by imitating a well-trained network, the tiny intermediate optical flow network has performance similar to that of the well-trained network with a small number of resources. λ_f is set to 0.01.

(3) Census loss

The census loss, L_c, is used to evaluate the illumination changes in the optical flow estimation network [3, 22] and to maintain the brightness constancy. The variable λ_c is set to 0.01.

(4) Perceptual loss

The perceptual loss, L_p, is a loss function for optimizing inference for learning through high-level features extracted from a pretrained network [10, 23]:

$$L_p = \|\phi_{VGG}(I_{GT}) - \phi_{VGG}(I_t)\|_2, \tag{7}$$

where ϕ denotes the part of the pretrained VGG-16 network trained for the ImageNet dataset. L_p can suppress the blurring that is likely to occur when utilizing the reconstruction loss. Thus, the detail of the predicted image is preserved and a sharper result is obtained. The variable λ_p is set to 0.005.

4 Experiments

4.1 Training Dataset and Benchmark Configuration

(1) Training Datasets

Previous studies on VFI [2, 6–8, 24] have adopted various training datasets. Among them, we trained the proposed network with the Vimeo90K triplet dataset, which is divided into training and testing sets. The Vimeo90K triplet dataset has 55,094 triplets of 448 $\times$ 256 images, of which 51,312 triplets were used for training. For network learning of the intermediate optical flow network, the optical flow dataset of the Vimeo90K triplet dataset was newly constructed using RAFT, an optical flow estimation network. The data augmentation technique used in [3] and the results of double-folding the Vimeo90K triplet dataset image size using bilinear interpolation were used to improve the learning of the intermediate optical flow network.

(2) Benchmark Dataset

The proposed method is evaluated with benchmark datasets like Middlebury, UCF101, Vimeo90K triplet, and Vimeo90K septuplet. UCF101 has been used commonly for VFI. This dataset consists of 256 $\times$ 256 resolution sequences. The Middlebury optical flow benchmark dataset has the motion of various objects and consists of various resolution sequences smaller than or equal to 720p. In the Vimeo90K triplet, there are 3,782 triplet sets with 448 $\times$ 256 resolution. The Vimeo90K septuplet was used in Section 3.3 to evaluate how to interpolate multiple frames with the one-shot structure, which has 7,824 sets consisting of 448 $\times$ 256 resolution sequences.

4.2 Comparisons with the State of the Art

We evaluated the inference speed and interpolation quality on various benchmark datasets in Table 2 and Figure 7. The methods listed below are typical networks used in VFI. The proposed methods have several network parameters but have faster interpolation speed than any of the listed methods. In addition, when loading a model to interpolate 720p resolution, only 2.2 gigabytes of graphics processing unit (GPU) memory are used, so the method has the advantage of running on a GPU with a small amount of memory. The overall interpolation quality exhibits higher performance than other proposed methods, and the quality performance is maintained even when the half-precision is applied. In addition, considering that the proposed method has

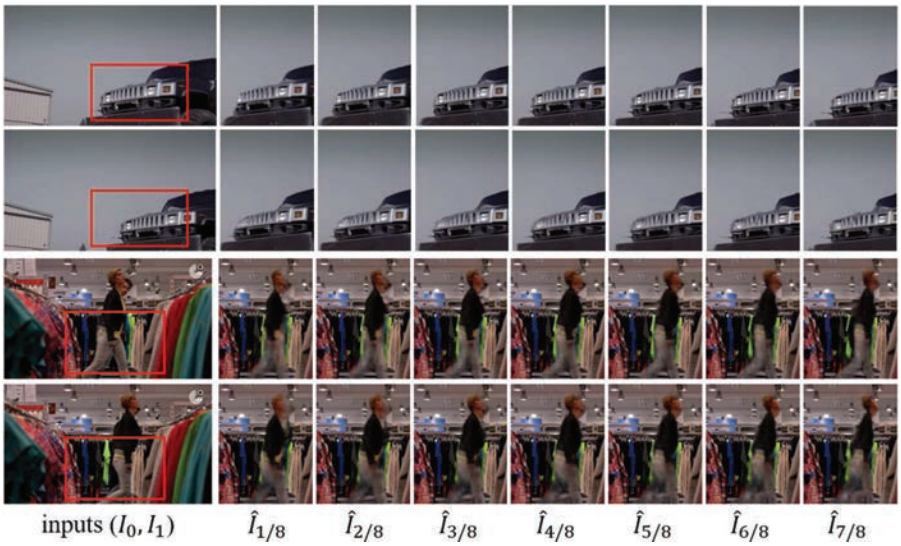

Figure 7 Visual comparisons with multiple frames between ours (first and third rows) and RIFE (second and forth rows) on the Vimeo90K triplet dataset.

higher accuracy than the RIFE learns under the same conditions, the predicted intermediate bidirectional optical flow is more accurately learned than the monodirectional flow.

4.3 Reduction of Inference Time

The half-precision implementation and one-shot structure to reduce inference time are analyzed. They can be applied independently without changing the network structure.

(1) Half-precision implementation

Table 2 presents the results of the half-precision implementation that cuts out lower 16 bits of the learned weight nodes in the Nvidia RTX 2080 TI. The FP16 precision results in a 21% faster speed than the FP32 precision, whereas the quality is slightly degraded in the peak signal-to-noise ratio (PSNR) by 0.04 dB on the Vimeo90K triplet dataset. For 480p sequence, the inference time of the proposed method with FP16 implementation is 11.9 ms and it is equal to 84 fps. This result indicates that the lower 16 bits of the 32 bits of the learned weight node do not significantly affect the interpolation

Table 2 Performance comparison on the UCF101, Vimeo90K triplet, and Middlebury datasets. Inference times are measured on RTX 2080 TI for 480p resolution

	Inference	Parameters	*Vimeo90K triplet*		*Middlebury*	*UCF101*	
Method	Time (ms)	(Million)	PSNR	SSIM	IE	PSNR	SSIM
Super-SloMo	34.1	19.8	33.15	0.966	2.28	34.75	0.968
SepConv	33.5	21.6	33.79	0.970	2.27	34.78	0.976
TOFlow	58.4	1.1	33.73	0.968	2.15	34.58	0.976
MEMC-Net	79.5	70.3	34.40	0.970	2.12	35.01	0.968
DAIN	82.1	24	34.71	0.976	2.04	35.00	0.968
CAIN	21.0	42.8	34.65	0.973	2.28	34.91	0.969
BMBC	505.9	11	35.01	0.976	2.04	35.15	0.969
RIFE	13.8	10.4	35.51	0.978	2.06	35.10	0.965
Ours (FP16)	11.9	14	35.54	0.978	2.09	35.11	0.966
Ours (FP32)	14.4	14	35.58	0.978	2.09	35.14	0.969

Table 3 Comparison of the one-shot structure

	Inference Time	*Vimeo90K septuplet*					
	(Three-Frame	PSNR			SSIM		
Method	Interpolation)	$I_{1/4}$	$I_{2/4}$	$I_{3/4}$	$I_{1/4}$	$I_{2/4}$	$I_{3/4}$
FP32	43.2	30.63	29.17	30.63	0.910	0.889	0.910
FP16	35.7	30.62	29.17	30.62	0.909	0.889	0.910
FP32 + one-shot	28.3	29.42	29.17	29.36	0.902	0.889	0.902
FP16 + one-shot	23.4	29.42	29.17	29.35	0.902	0.889	0.902

quality and the proposed method can improve the inference speed significantly.

(2) One-shot structure

Table 3 lists the results of applying the one-shot structure along with the half-precision implementation on the Vimeo90K septuplet dataset. The experimental results were taken from the first and fifth frames in the septuplet dataset as input (I_0, I_1), and three frames of consecutive time $(I_{1/4}, I_{2/4}, I_{3/4})$ were interpolated and compared with the ground truth. As a result of the experiment, the interpolation quality decreased by about 0.8 dB on average compared to the recursive structure like RIFE, but the speed was reduced by up to 52.6%. This result reveals that the proposed one-shot structure reduces the computational complexity at a slight cost of the interpolation quality. As a result, the inference works in real time through the trade-off of the speed and interpolation quality.

Table 4 Quality comparison of various loss functions

	Vimeo90K triplet	
Combination of Loss Functions	PSNR	SSIM
$L_r + L_c$	35.22	0.976
$L_r + L_c + L_f$	35.53	0.978
$L_r + L_c + L_f + L_p$	35.58	0.978

Table 5 Quality comparison of the IFNet convolutional depth

			Vimeo90K triplet	
Depth	Parameters (Million)	Inference Time	PSNR	SSIM
$\times 1$	10.4	13.8	35.52	0.978
$\times 1.25$	14.0	14.4	35.58	0.978
$\times 1.5$	18.3	16.3	35.57	0.978

4.4 Ablation Study

We changed the network hyperparameters and analyzed the parameters that derive the optimal inference results. These experiments were trained [25–29] and evaluated on the Vimeo90K triplet dataset.

Table 4 is the result of comparing the quality when each loss function was applied. For comparison, the perceptual loss L_p and optical flow loss L_f were applied and compared, based on the model trained by applying only the loss functions L_r and L_c. As a result, when the optical flow was directly learned, a large quality improvement of 0.31 dB occurred, and the perceptual loss also affected the performance improvement.

Table 5 is the result according to the number of output channels of the convolutional layer included in IFNet, an intermediate optical flow estimation network. The number of initial channels is from RIFE. The results reveal that increasing the depth improves the interpolation quality, but increasing the depth over a certain level has no effect. The depth with optimal interpolation quality was selected through a trade-off of about 5% interpolation speed.

Finally, Table 6 presents the results of the quality comparison according to the ground truth and interpolation method used for learning the optical flow network. Moreover, $F_{\times n}$ is the result of learning with the optical flow prediction output bilinearly interpolated at *n* times the size of the frame for training for the L_f loss function and is used when interpolating the optical flow for a coarse-to-fine structure. Thus, when learning by increasing the optical flow, a more sophisticated optical flow and enhanced results were obtained. The result of interpolation using pixel-area relation was higher than the result of down-sampling using simple bilinear interpolation, confirming

Table 6 Quality comparison of combination of two methods, the former is a ground truth generation method of optical flow and the latter is a down-sampling method

	Vimeo90K triplet	
Method	PSNR	SSIM
$F_{\times 1}$+ bilinear	35.44	0.977
$F_{\times 2}$+ bilinear	35.55	0.978
$F_{\times 1}$+ area	35.45	0.977
$F_{\times 2}$+ area	35.58	0.978

high results. In general, the pixel-area relation best preserves the quality when resizing at a lower resolution [30], and it has been confirmed that it is also applied to the optical flow.

5 Conclusion

This paper proposes a fast VFI method. It improves interpolation frame quality by replacing monodirectional optical flow prediction of RIFE with the bidirectional optical flow prediction, and it reduces runtime using the one-shot structure and the half-precision implementation. The cognitive prediction accuracy is more improved by applying the perceptual loss function. The experimental results showed the high interpolation quality of 35.54 dB on the datasets Vimeo90K and the real-time speed of 84 fps for 480p resolution. If optical flow is estimated more precisely, the interpolation quality can increase. Therefore, optical flow prediction network like IFNet needs to be researched further.

Acknowledgement

This research was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-00452, Development of Adaptive Viewer-centric Point cloud AR/VR(AVPA) Streaming Platform) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2020R1F1A1070302).

References

[1] S. Niklaus, L. Mai, F. Liu, Video frame interpolation via adaptive separable convolution, In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

[2] X. Cheng, Z. Chen, Video frame interpolation via deformable separable convolution, In Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI), 2020.
[3] Z. Huang, T. Zhang, W. Heng, B. Shi, S. Zhou, RIFE: Real-Time Intermediate Flow Estimation for Video Frame Interpolation, arXiv preprint arXiv:2011.06294v2, 2021.
[4] B. Choi, S. Lee, S. Ko, New frame rate up-conversion using bi-directional motion estimation, IEEE Transactions on Consumer Electronics, 2000.
[5] W. Bao, W. Lai, C. Ma, X. Zhang, Z. Gao, M. Yang, Depth-aware video frame interpolation, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[6] X. Cheng, Z. Chen, Memc-net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement, IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI), 2018.
[7] S. Niklaus, F. Liu, Context-aware synthesis for video frame interpolation, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
[8] S. Niklaus, L. Mai, F. Liu, Video frame interpolation via adaptive convolution In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[9] O. Ronneberger, P. Fischer, T. Brox, Convolutional Networks for Biomedical Image Segmentation, In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015.
[10] H. Jiang, D. Sun, V. Jampani, M. Yang, E. Miller, J. Kautz, Super slomo: High quality estimation of multiple intermediate frames for video interpolation, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
[11] T. Hui, X. Tang, C. Loy, Liteflownet: A lightweight convolutional neural network for optical flow estimation, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
[12] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Smagt, D. Cremers, T. Brox, Flownet: Learning optical flow with convolutional networks, In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015.

[13] Y. Liu, L. Xie, L. Siyao, W. Sun, Y. Qiao, C. Dong, Enhanced quadratic video interpolation, In Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[14] Y. Liu, L. Xie, L. Siyao, W. Sun, Y. Qiao, C. Dong, Enhanced Quadratic Video Interpolation European Conference on Computer Vision (ECCV), 2020.

[15] S. Niklaus, F. Liu, Softmax Splatting for Video Frame Interpolation, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[16] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[17] W. Shi, J. Caballero, F. Huszar, J. Totz, A. Aitken, R. Bishop, D. Ruechert, Z. Wang, Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[18] M. Choi, H. Kim, B. Han, N. Xu, K. Lee, Channel attention is all you need for video frame interpolation, In Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI), 2020.

[19] F. Reda, D. Sun, A. Dundar, M. Shoeybi, G. Liu, Unsupervised Video Interpolation Using Cycle Consistency, In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2019.

[20] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox, Flownet 2.0: Evolution of optical flow estimation with deep networks, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[21] Z. Teed, J. Deng, Raft: Recurrent all-pairs field transforms for optical flow, In Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[22] S. Meister, J. Hur, S. Roth, UnFlow: Unsupervised learning of optical flow with a bidirectional census loss, In Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI), 2018.

[23] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, In Proceedings of the European Conference on Computer Vision (ECCV), 2016.

[24] S. Meyer, O. Wang, H. Zimmer, M. Grosse, A. Sorkine-Hornung, Phase-based frame interpolation for video, In Proceedings of the IEEE

Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[25] A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861, 2017.
[26] I. Loshchilov, F. Hutter, Fixing weight decay regularization in adam, International Conference on Learning Representations (ICLR), 2018.
[27] S. Kaufman, S. Rosset, C. Perlich, O. Stitelman, Leakage in data mining: Formulation, detection, and avoidance, ACM Transactions on Knowledge Discovery from Data (TKDD), 2012.
[28] U. Ninrutsirikun, D. Pal, C. Arpnikanondtand, B. Watanapa, Unified Model for Learning Style Recommendation, Journal of Web Engineering (JWE), 2021.
[29] J. G. Enríquez, A. Martínez-Rojas, D. Lizcano, A. Jiménez-Ramírez, A Unified Model Representation of Machine Learning Knowledge, Journal of Web Engineering (JWE), 2020.
[30] G. Sun, J. Zhang, K. Zheng, X. Fu, Eye Tracking and ROI Detection within a Computer Screen Using a Monocular Camera, Journal of Web Engineering (JWE), 2020.

Biographies

Minseop Kim received a B.S. in the department of information and communication engineering from Hanbat National University, Daejeon, Korea, in 2018, where he received a M.S in the department of multimedia engineering. Currently, he is working toward Ph.D. degree in the department of multimedia engineering in Hanbat National University. His research interests include computer vision, machine learning, and parallel processing.

Haechul Choi received a B.S. in electronics engineering from Kyungpook National University, Daegu, Korea, in 1997. He received his M.S. and Ph.D. in electrical engineering from Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 1999. He was a senior researcher at the Broadcasting Media Research Department of Electronics and Telecommunications Research Institute until 2010 and was an adjunct professor at the University of Science and Technology. He was a visiting professor in Florida Institute Technology from 2015 to 2016. He is currently a professor at Hanbat National University. His current research areas include image processing, image compression, video coding, video compression, signal processing, computer vision, deep learning.