Meta-Interpolation: Time-Arbitrary Frame Interpolation via Dual Meta-Learning

Shixing Yu[†], Yiyang Ma[†], Wenhan Yang[†], Wei Xiang[‡], Jiaying Liu[†]*
[†]Wangxuan Institute of Computer Technology, Peking University, Beijing, China
[‡]Bigo, Beijing, China

***Abstract*—Existing video frame interpolation methods can only interpolate the frame at a given intermediate time-step, *e.g.* 1/2. In this paper, we aim to explore a more generalized kind of video frame interpolation, that at an arbitrary time-step. To this end, we consider processing different time-steps with adaptively generated convolutional kernels in a unified way with the help of meta-learning. Specifically, we develop a dual meta-learned frame interpolation framework to synthesize intermediate frames with the guidance of context information and optical flow as well as taking the time-step as side information. First, a content-aware meta-learned flow refinement module is built to improve the accuracy of the optical flow estimation based on the down-sampled version of the input frames. Second, with the refined optical flow and the time-step as the input, a motion-aware meta-learned frame interpolation module generates the convolutional kernels for every pixel used in the convolution operations on the feature map of the coarse warped version of the input frames to generate the predicted frame. Extensive qualitative and quantitative evaluations, as well as ablation studies, demonstrate that, via introducing meta-learning in our framework in such a well-designed way, our method not only achieves superior performance to state-of-the-art frame interpolation approaches but also owns an extended capacity to support the interpolation at an arbitrary time-step.**

I. Introduction

Video frame interpolation creates non-existent intermediate frames of the input video and maintains the newly generated video to be continuous spatially and temporally and to have a pleasing visual effect. The technique has been studied widely and becomes a hot research topic in the video processing community. Its applications range from frame rate up conversion [1], [2], novel view synthesis [3], and inter prediction in video coding [4], [5].

Conventional frame interpolation estimates the optical flow of the input frames first, then infer the optical flow at the intermediate time-step, and finally warp the input pixels to the target ones under the guidance of the optical flow [6], [7]. This kind of method heavily relies on optical flow estimation, therefore their performance is unstable if there are large motions as the optical flow estimation is usually inaccurate in this case. Furthermore, the conventional optical flow estimation might be time-consuming, therefore the complexity of these methods is usually high.

Nowadays, convolutional neural networks (CNN) have been applied to synthesizing the intermediate frames, achieving promising performance in visual quality and time efficiency. All methods can be classified into three branches: 1) direct generation [8] that takes the original frames as input and directly predicts the intermediate frames; 2) flow-guided method [9]–[12] that simulates the align-and-synthesis paradigm; 3) adaptive kernel based method [13]–[15] that adopts a flow-free pipeline, where the convolution kernels are learned by passing the original frames through a CNN.

All previous methods come across several neglected issues. 1) It is a dilemma whether to adopt the optical flow or not. Optical flow estimation is an effective representation of motion modeling. However, its estimation is not robust when large motions are included. 2) Due to the fixed model parameters, all methods can only be applied to handling the interpolation at the given time-step adopted in the training stage, *e.g.* 1/2. 3) As all modules adopt fixed parameters, their adaptivity is not enough to handle different contents and motion conditions accurately and robustly.

Recently, meta-learning, is introduced to increase the model's adaptivity via adjusting the model based on the testing conditions and content of the input images/videos for many computer vision and image processing tasks [16]–[20]. These works motivate us to address the above-mentioned issues via meta-learning.

In our work, we still follow the paradigm of align-and-synthesis and aim to realize the time-arbitrary video frame interpolation with accurate and robust modeling of motions. First, to achieve the time-arbitrary video frame interpolation, we build a meta-learned frame interpolation module that takes both the optical flow and time-step as the input for generating the convolutional kernels used in the convolution operations adaptively to produce the predicted intermediate frame. Besides, it also makes our model more adaptive to different motion contexts and contents when meta-learned adaptive convolutions are introduced. Second, a meta-learned flow refinement module is introduced to improve the accuracy of the optical flow estimation based on the down-sampled version of the input frames. As shown in the visual results and the quantitative results, the proposed method outperforms state-of-the-art methods and can provide a better solution for the arbitrary-time video frame interpolation.

*Corresponding author.

This work is supported by the National Key Research and Development Program of China under Grant No. 2018AAA0102702, the National Natural Science Foundation of China under Contract No.62172020, State Key Laboratory of Media Convergence Production Technology and Systems, and Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).

978-1-6654-8485-5/22/$31.00 ©2022 IEEE

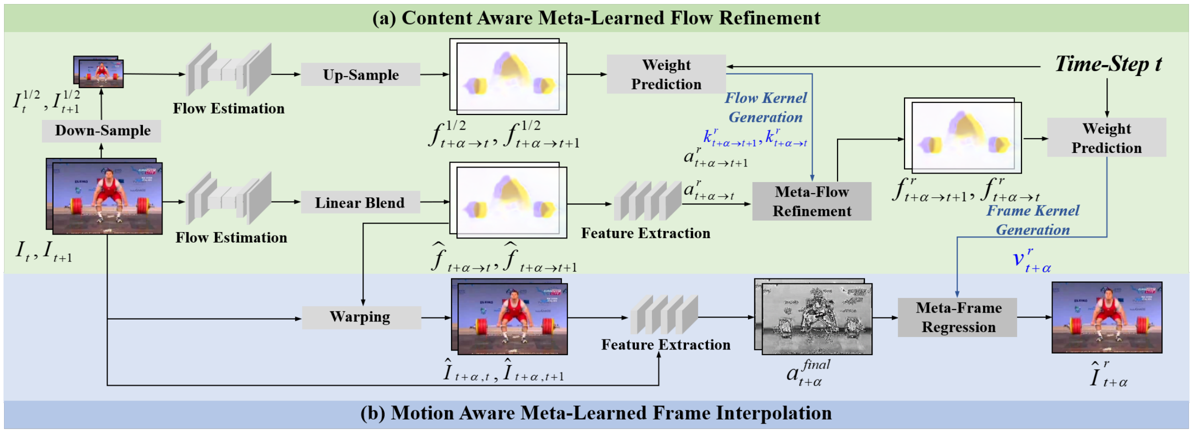
(a) Content Aware Meta-Learned Flow Refinement
$I_t^{1/2}, I_{t+1}^{1/2}$
Down-Sample
Flow Estimation
Up-Sample
$f^{1/2}_{t+\alpha\to t}, f^{1/2}_{t+\alpha\to t+1}$
Weight Prediction
Time-Step t
Flow Kernel Generation
$k^r_{t+\alpha\to t+1}, k^r_{t+\alpha\to t}$
$a^r_{t+\alpha\to t+1}$
$a^r_{t+\alpha\to t}$
Meta-Flow Refinement
$f^r_{t+\alpha\to t+1}, f^r_{t+\alpha\to t}$
Weight Prediction
Frame Kernel Generation
$v^r_{t+\alpha}$
I_t, I_{t+1}
Flow Estimation
Linear Blend
$\hat{f}_{t+\alpha\to t}, \hat{f}_{t+\alpha\to t+1}$
Feature Extraction
Warping
$\hat{I}_{t+\alpha,t}, \hat{I}_{t+\alpha,t+1}$
Feature Extraction
$a^{final}_{t+\alpha}$
Meta-Frame Regression
$\hat{I}^r_{t+\alpha}$
(b) Motion Aware Meta-Learned Frame Interpolation

Fig. 1. The framework of our proposed method. (a) The content-aware meta-learned flow refinement module improves the accuracy of the optical flow estimation based on the down-sampled version of the input frames. (b) With the refined optical flow and the time-step as the input, the motion-aware meta-learned frame interpolation module generates the pixel-wise convolutional kernels, used to fuse the coarse warped version of the input frames for the frame interpolation.

II. Meta-Interpolation

A. Motivations

To address the issues mentioned in introduction, we have the following three motivations in our minds to guide our architecture design:

- *Improving robustness and accuracy of motion estimation*. Our architecture inherits the align-and-synthesis paradigm due to the excellent performance this kind of method offers. Furthermore, we hope to inject the motion awareness mechanism into the motion estimation.
- *Providing feasibility to decide the time-step of the generated intermediate frame in a flexible way*. It is generally preferred if the framework can take the time-step as the input to support time-arbitrary interpolation under the control of an input time-step.
- *Improving adaptivity by making the developed architecture more aware of content and motion contexts.* Thus, in our architecture, we hope to integrate time-step utilization, motion estimation and context perception jointly.

B. Arbitrary-Time Interpolation

Conventional video frame interpolation techniques usually take two adjacent frames I_t and I_{t+1} as input and generate the intermediate frame at a given time-step, usually $I_{t+0.5}$. The process $\Phi_f(\cdot)$ can be formulated as follows,

$$I_{t+0.5} = \Phi_{fixed}(I_t, I_{t+1}|\theta_f), \tag{1}$$

where θ_f is the model parameter and $fixed$ means the fixed time-step. For most of the existing deep-learning video frame interpolation methods, they can only predict the target frame at that time-step.

For the time-arbitrary video frame interpolation problem, the time-step is relaxed as a controllable input, instead of a fixed number, which can be adjusted freely in the testing phase. Eqn. (1) can be extended as follows,

$$I_{t+\alpha} = \Phi_{arbitrary}(I_t, I_{t+1}, \alpha|\theta_a),\ 0 < \alpha < 1, \tag{2}$$

where θ_a is the model parameter and $arbitrary$ means the arbitrary time-step. Many traditional interpolation methods, can be applied to generate the intermediate frame at an arbitrary time-step. In our work, we hope to give deep models such capacity via exploiting the power of meta-learning. That is, the model parameters used for interpolation can be dynamically decided in the testing phase based on the input context, *e.g.* frame context, optical flow, and time-step.

C. Framework Overview

The architecture of our framework is briefly shown in Fig. 1. In general, the proposed method views video frame interpolation as the result of a convolutional neural network, adaptive instead of fixed based on the testing context.

Specifically, we develop a dual meta-learned frame interpolation framework to synthesize the intermediate frame $I_{t+\alpha}$ with the guidance of optical flow $f_{t+\alpha,t}$ and $f_{t+\alpha,t+1}$ as well as taking the time-step α as side information. The total pipeline consists of a *content-aware meta-learned flow refinement module* and a *motion-aware meta-learned frame interpolation module*. In the next part, these two modules will be further introduced in detail.

D. Content-Aware Meta-Learned Flow Refinement

1) Initial Optical Flow Estimation. We first aim to obtain a good motion estimation, fully considering the content information and the time-step. To this end, we construct a coarse-to-fine optical flow estimation and refinement pipeline. We first estimate the initial optical flow at both the original resolution and half resolution spaces. The flow estimation process is denoted by $E(\cdot)$, then we obtain the optical flow estimations as follows:

$$\begin{aligned} f^{1/2}_{t\to t+1} &= U\left(E\left(I_t^{1/2}, I_{t+1}^{1/2}\right)\right), \\ f^{1/2}_{t+1\to t} &= U\left(E\left(I_{t+1}^{1/2}, I_t^{1/2}\right)\right), \end{aligned} \tag{3}$$

$$\begin{aligned} f_{t\to t+1} &= E\left(I_t, I_{t+1}\right), \\ f_{t+1\to t} &= E\left(I_{t+1}, I_t\right), \end{aligned} \tag{4}$$

where $U(\cdot)$ is the up-sampling process that projects the frame/flow back to the original resolution space.

2) Linear Blend of Flows. Then, based on [21], the flow related to the time-step α can be inferred by the linear blend:

$$\begin{aligned} \hat{f}_{t+\alpha\to t} &= -(1-\alpha)\alpha f_{t\to t+1} + \alpha^2 f_{t+1\to t}, \\ \hat{f}_{t+\alpha\to t+1} &= (1-\alpha)^2 f_{t\to t+1} - \alpha(1-\alpha) f_{t+1\to t}. \end{aligned} \tag{5}$$

3) Flow Kernel Generation. After that, we refine the flow estimation based on these initial estimations. In our method, following [21], we seek to utilize the multi-scale information to realize the optical flow information refinement.

The estimated optical flows obtained from $f^{1/2}_{t\to t+1}$ and $f^{1/2}_{t+1\to t}$ are utilized to provide the context information to infer the refined version of optical flows at the original resolution space. Through a cascaded convolutional and fully connected layers, for each location (i,j), input tensors $s^r_{t+\alpha\to t}, s^r_{t+\alpha\to t+1} \in R^{6\times hw}$ (the superscript r denoting refinement) with both optical flows and time-step information is collected in the following way:

$$\begin{aligned} s^r_{t+\alpha\to t}(i,j) = s^r_{t+\alpha\to t+1}(i,j) = \\ (f^{1/2}_{t\to t+1}(i,j), f^{1/2}_{t+1\to t}(i,j), \alpha, 1-\alpha). \end{aligned} \tag{6}$$

After that, the refined side information matrices $s^r_{t+\alpha\to t}$ and $s^r_{t+\alpha\to t+1}$ are calculated and feed-forwarded to the given network $K(\cdot)$ to derive the convolutional kernels $k^r_{t+\alpha\to t}$ and $k^r_{t+\alpha\to t+1}$ as follows,

$$k^r_{t+\alpha\to t} = K\left(s^r_{t+\alpha\to t}\right), k^r_{t+\alpha\to t+1} = K\left(s^r_{t+\alpha\to t+1}\right). \tag{7}$$

4) Feature Extraction. Different from previous methods, in our method, the convolution is not operated directly on the input images. Our adaptive convolution operations are applied to the extracted features $a^r_{t+\alpha\to t}$, which are generated via the cascaded convolutional layers $A(\cdot)$ as follows,

$$a^r_{t+\alpha\to t} = A\left(\hat{f}_{t+\alpha\to t}\right). \tag{8}$$

5) Meta-Flow Refinement. Finally, the refined optical flows are inferred as follows:

$$f^r_{t+\alpha\to t} = a^r_{t+\alpha\to t} \otimes k^r_{t+\alpha\to t}, \tag{9}$$

where $\otimes$ denotes the convolutional operation.

E. Motion-Aware Meta-Learned Frame Prediction

Based on guidance of refined optical flows, we perform the video frame interpolation.

1) Initial Frame Warping. Based on the initial estimated flows, we can obtain the coarsely warped results as follows,

$$\begin{aligned} \hat{I}_{t+\alpha,t} &= W(I_t, f^r_{t+\alpha\to t}), \\ \hat{I}_{t+\alpha,t+1} &= W(I_{t+1}, f^r_{t+\alpha\to t+1}), \end{aligned} \tag{10}$$

where $W(\cdot)$ is backward warping function, which can be implemented by bilinear interpolation and is differentiable [10].

2) Frame Kernel Generation. We use optical flow information and time stamp as side information to predict the convolutional filters. For each spatial location (i,j), we create the input tensor as follows:

$$n^r_{t+\alpha}(i,j) = (f^r_{t+\alpha\to t}(i,j), f^r_{t+\alpha\to t+1}(i,j), \alpha, 1-\alpha), \tag{11}$$

where $f^r_{t+\alpha\to t}(i,j)$ is the refined optical flow extracted from $I_{t+\alpha}$ to I_t at location (i,j) including two flow fields. The superscripted f^r which denotes the optical flow is a refined version generated from the motion-aware meta-learned optical flow refinement in Sec. 3.4.

After that, the refined side information matrice $n^r_{t+\alpha}$ is feed-forwarded to the given network $V(\cdot)$ to derive the convolutional kernels $v^r_{t+\alpha}$ as follows,

$$v^r_{t+\alpha} = V\left(n^r_{t+\alpha}\right). \tag{12}$$

3) Meta-Frame Regression. To present a coarse-to-fine process, we concatenate the initial warped frames $\hat{I}_{t+\alpha,t}$ and $\hat{I}_{t+\alpha,t+1}$ together with the original input I_t, I_{t+1} and extract features through a cascaded convolutional layer $A_{final}(\cdot)$ similar to the feature extraction layer in Sec 3.4 and generate feature maps as follows:

$$a^{final}_{t+\alpha} = A_{final}\left(\hat{I}_{t+\alpha,t}, \hat{I}_{t+\alpha,t+1}, I_t, I_{t+1}\right). \tag{13}$$

Finally, we infer the video frame interpolation result as follows:

$$\hat{I}^r_{t+\alpha} = a^{final}_{t+\alpha} \otimes v^r_{t+\alpha}. \tag{14}$$

where $\otimes$ denotes the convolutional operation.

F. Implementation Details

1) Architecture Details. We choose residual dense network (RDN) [22] as the feature extraction network. For frame kernel generation, the network consists of 12 residual dense blocks. Each RDB consists of 8 convolution layers and 64 channels. In the flow kernel generation stage, the network's corresponding hyper-parameters are 4, 4, 32. The weight prediction networks for both flow refinement and frame interpolation consist of 2 convolution layers, 2 fully connected layers. We use the state-of-the-art optical flow prediction algorithm PWC-Net [23] for optical flow estimation.

2) Loss Function. We denote the ground-truth frame as $I_{t+\alpha}$ and our prediction as $\hat{I}^r_{t+\alpha}$. The reconstruction loss is defined as follows:

$$L_r = \left\|\hat{I}^r_{t+\alpha} - I_{t+\alpha}\right\|_1. \tag{15}$$

For optical flow refinement, we introduce three kinds of losses: flow loss L_f, warping loss L_w, and smooth loss L_s. The flow loss L_f plays a role of regularization for the optical flow refinement by enforcing the consistency between the refined optical flow and the original optical flow (estimated directly by a pretrained model). It is denoted as follows:

$$L_f = \left\|f^r_{t+\alpha\to t} - f^p_{t+\alpha\to t}\right\|_1 + \left\|f^r_{t+\alpha\to t+1} - f^p_{t+\alpha\to t+1}\right\|_1. \tag{16}$$

The warping loss L_w regularizes the consistency between the warped frame and adjacent frame, implicitly instructing the refinement of the optical flow. It is represented as follows,

$$\begin{aligned} L_w = &\left\|W\left(I_t, f^p_{t+\alpha\to t}\right) - W\left(I_t, f^r_{t+\alpha\to t}\right)\right\|_1 + \\ &\left\|W\left(I_{t+1}, f^p_{t+\alpha\to t+1}\right) - W\left(I_t, f^r_{t+\alpha\to t+1}\right)\right\|_1. \end{aligned} \tag{17}$$

The commonly used smoothness loss L_s is also adopted to suppress the artifacts in the generated optical flow estimations as follows,

$$L_s = \left\|\Delta\left(f^r_{t+\alpha\to t}\right)\right\|_1 + \left\|\Delta\left(f^r_{t+\alpha\to t+1}\right)\right\|_1, \tag{18}$$

where $\Delta(\cdot)$ is the gradient operator.

In summary, our total training loss is given as follows,

$$L = \lambda_r L_r + \lambda_f L_f + \lambda_w L_w + \lambda_s L_s, \tag{19}$$

where λ_r, λ_f, λ_w and λ_s are the weighting parameters that balance the importance of each term. We empirically set $\lambda_r = 1.0$ $\lambda_f = 0.02$, $\lambda_w = 0.2$ and $\lambda_s = 0.5$.

TABLE I
QUANTITATIVE EVALUATION ON THE UCF-101, VIMEO90K AND MIDDLEBURY DATASETS. THE BEST RESULTS ARE DENOTED IN RED WHILE THE SECOND BEST RESULTS ARE DENOTED IN BLUE.

Network	Vimeo90K		UCF-101		MiddleBury
	PSNR	SSIM	PSNR	SSIM	IE
MIND [24]	33.50	0.9429	33.93	0.9661	3.35
DVF [9]	31.54	0.9462	34.12	0.9631	7.75
SepConv [15]	33.79	0.9702	34.69	0.9669	2.28
SuperSlomo [10]	33.53	0.9653	34.26	0.9664	2.47
CyclicGen [25]	32.10	0.9492	35.11	0.9681	2.86
CAIN [26]	34.40	0.9715	34.87	0.9685	2.28
DAIN [14]	34.70	0.9756	35.00	0.9683	2.04
MIN	34.80	0.9761	35.05	0.9687	2.08

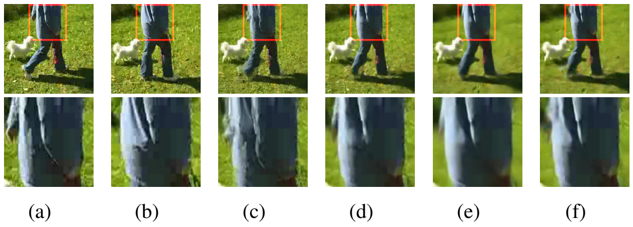

Fig. 2. Visual result of the video frame interpolation by different methods. (a) and (b): original inputs I_0 and I_1. (c)→(e): the results produced by SepConv [15], DAIN [14], CAIN [26], respectively. (f): the result of our method (MIN).

III. EXPERIMENTS

1) Datasets and Metrics. We train our proposed video frame interpolation method on Vimeo90K dataset [12] and validate on UCF101 [27], Vimeo90K [12] and Middlebury benchmark [6]. Following the setting in [6], We also report Interpolation Error (IE) on Middlebury benchmark.

2) Training Strategies and Hyper-Parameter Setting. We train the network for 20 epochs with a mini-batch size of 2. The initial learning rate is set to 0.001 and a reduce on plateau strategy. Adam [28] optimzer is used to update the network parameters, with $\beta_1 = 0.9, \beta_2 = 0.999$.

3) Evaluation. We compare the performance of our approach against several SotA methods on quantity and quality. The results are shown in Table I and Fig. 2. Our method is denoted as MIN. As we can see in Table I, our method outperforms almost all the state-of-the-art methods in all metrics.

4) Ablation Studies. We analyze the effectiveness of each component and constraint of our method in Table II.

- *Motion-Aware Meta-Learned Frame Prediction.* We denote **MIN-Base** as the model directly feed-forwarding the two concatenated coarse generation results through the RDN. Meanwhile, we also compare the result of another kernel-based method, SepConv [15].

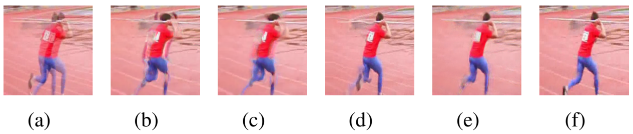

Fig. 3. Visual comparison of the proposed method with different components. (a): Overlapped input. (b): **MIN-Base**: directly feed-forwarding the input through an RDN. (c): **MIN-UNR**: MIN-Base + motion-aware meta-learned model. (d): SepConv [15]. (e): **MIN-Full**: our full version. (f): Ground truth.

TABLE II
ABLATION STUDY FOR THE MOTION-AWARE META-LEARNED FRAME PREDICTION AND THE CONTENT-AWARE META-LEARNED OPTICAL FLOW REFINEMENT. THE BEST RESULTS ARE DENOTED IN RED.

Network	Metrics	Vimeo	UCF-101
SepConv [15]	PSNR	33.79	34.69
	SSIM	0.9702	0.9669
MIN-Base	PSNR	34.30	34.49
	SSIM	0.9735	0.9670
MIN-UNR	PSNR	34.47	34.61
	SSIM	0.9746	0.9674
MIN-UNC	PSNR	34.62	34.92
	SSIM	0.9750	0.9682
MIN-Full	PSNR	34.80	35.05
	SSIM	0.9761	0.9687

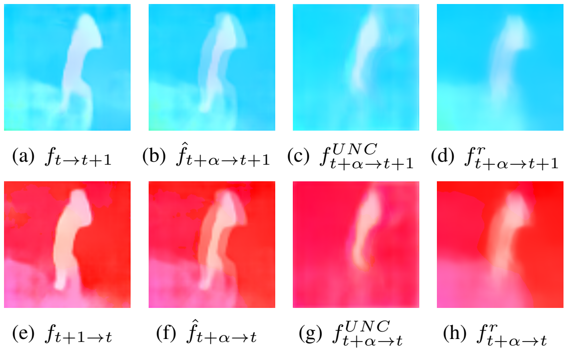
(a) $f_{t\to t+1}$
(b) $\hat{f}_{t+\alpha\to t+1}$
(c) $f^{UNC}_{t+\alpha\to t+1}$
(d) $f^{r}_{t+\alpha\to t+1}$
(e) $f_{t+1\to t}$
(f) $\hat{f}_{t+\alpha\to t}$
(g) $f^{UNC}_{t+\alpha\to t}$
(h) $f^{r}_{t+\alpha\to t}$

Fig. 4. Visualization of the optical flow results of different versions of our method.

- *Content-Aware Meta-Learned Flow Refinement.* Then, we perform the ablation study on the content-aware meta-learned flow refinement module. We denote the model without the proposed optical flow refinement module as **MIN-UNR**, the model that only uses the reconstruction loss L_r in the training loss as **MIN-UNC**. Besides, we visualize the optical flow to show the effectiveness of this module in Fig. 4. In general, both of our last two versions can well capture the motions of foreground objects.
- *Visual Comparisons.* We show qualitative results on one image selected from UCF101 in Fig. 3. Our method successfully generates spatially consistent result in Fig. 3 (e).

IV. CONCLUSIONS

In this paper, we develop a dual meta-learned frame interpolation framework that is capable to synthesize the intermediate frame at an arbitrary intermediate time-step. First, we create a content-aware meta-learned flow refinement module that takes the down-sampled version of the input frames as input to improve the accuracy and robustness of the optical flow estimation. Second, a motion-aware meta-learned frame interpolation module generates the convolutional kernels to generate the interpolated frame based on the refined optical flows and the time-step. Extensive qualitative and quantitative evaluations demonstrate the superiority of our method and the effectiveness of each component of our method.

References

[1] X. Hoangvan, J. Ascenso, and F. Pereira, "Improved matching criterion for frame rate upconversion with trilateral filtering," *Electronics Letters*, vol. 49, no. 2, 2013.

[2] H. S. Jung, U. S. Kim, and M. H. Sunwoo, "Simplified frame rate upconversion algorithm with low computational complexity," in *European Signal Processing Conference (EUSIPCO)*, 2012.

[3] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deep stereo: Learning to predict new views from the world's imagery," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] S. Xia, W. Yang, Y. Hu, and J. Liu, "Deep inter prediction via pixel-wise motion oriented reference generation," in *IEEE International Conference on Image Processing (ICIP)*, 2019.

[5] X. Huang, L. L. Rakêt, H. Van Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis transform domain wyner-ziv video coding including optical flow," in *IEEE International Workshop on Multimedia Signal Processing*, 2011.

[6] S. Baker, S. Roth, D. Scharstein, M. J. Black, J. P. Lewis, and R. Szeliski, "A database and evaluation methodology for optical flow," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[7] M. Werlberger, T. Pock, M. Unger, and H. Bischof, "Optical flow guided TV-L1 video interpolation and restoration," in *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, 2011.

[8] G. Long, L. Kneip, J. M. Alvarez, and H. Li, "Learning image matching by simply watching video," in *European Conference on Computer Vision (ECCV)*, 2016.

[9] L. Ziwei, R. Yeh, T. Xiaoou, L. Yiming, and A. Agarwala, "Video frame synthesis using deep voxel flow," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[10] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super slomo: High quality estimation of multiple intermediate frames for video interpolation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[11] S. Niklaus and F. Liu, "Context-aware synthesis for video frame interpolation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[12] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," *International Journal of Computer Vision*, vol. 127, no. 8, 2019.

[13] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[14] W. Bao, W.-S. Lai, C. Ma, X. Zhang, Z. Gao, and M.-H. Yang, "Depth-aware video frame interpolation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[16] S. Bell-Kligler, A. Shocher, and M. Irani, "Blind super-resolution kernel estimation using an internal-gan," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[17] A. Shocher, N. Cohen, and M. Irani, "“zero-shot” super-resolution using deep internal learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[18] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-sr: A magnification-arbitrary network for super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[19] S. Park, J. Yoo, D. Cho, J. Kim, and T. H. Kim, "Fast Adaptation to Super-Resolution Networks via Meta-Learning," *arXiv e-prints*, 2020.

[20] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *the Internatioanal Conference on Machine Learning (ICML)*, 2017.

[21] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[22] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[23] Z. Ren, O. Gallo, D. Sun, M.-H. Yang, E. B. Sudderth, and J. Kautz, "A fusion approach for multi-frame optical flow estimation," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019.

[24] G. Long, L. Kneip, J. M. Alvarez, H. Li, X. Zhang, and Q. Yu, "Learning image matching by simply watching video," in *European Conference on Computer Vision (ECCV)*, 2016.

[25] Y.-L. Liu, Y.-T. Liao, Y.-Y. Lin, and Y.-Y. Chuang, "Deep video frame interpolation using cyclic frame generation," in *Conference on Artificial Intelligence (AAAI)*, 2019.

[26] M. Choi, H. Kim, B. Han, N. Xu, and K. M. Lee, "Channel attention is all you need for video frame interpolation," in *Conference on Artificial Intelligence (AAAI)*, 2020.

[27] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *arXiv e-prints*, 2012.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference for Learning Representations (ICLR)*, 2015.