Alignment-aware Patch-level Routing for Dynamic Video Frame Interpolation (Supplementary Material)

Ban Chen[1]
ban.chen@samsung.com
Xin Jin[1]
jin.xin@samsung.com
Longhai Wu[1]
longhai.wu@samsung.com
Jie Chen[1]
ada.chen@samsung.com
Ilhyun Cho[2]
ih429.cho@samsung.com
Cheul-Hee Hahm[2]
chhahm@samsung.com

[1] Samsung Electronics (China) R&D Centre,
Nanjing
[2] Samsung Electronics,
South Korea

In this supplementary, we first provide more ablation studies in Sec. **??**. Then, more visualizations of comparison results and predicted route map are shown in Sec. **??**

1 Ablation Study

Design of ContextNet. To explore down-sampling scale of pyramid architecture within ContextNet, we try two designs of ContextNet. As shown in Fig. **??**, ContextNet v1 (our default choice) outputs the same original resolution features for each level, using 4 convolution layers without down-sampling. ContextNet v2 generates pyramid features through 4 convolution blocks with $[2,2,4,4]$ downscale. Tab. **??** shows ContextNet v1 helps static VFI(w/o APR) achieves the best accuracy, but cannot maintain competitive performance on X-2K under dynamic computation. We analysis that APR skip some important regions for X-2K interpolation.

model	ContextNet	X-2K	X-"4K"
Ours (w/o APR)	ContextNet v1 (default)	36.67/0.967	34.02/0.946
	ContextNet v2	36.61/0.967	33.94/0.946
Ours (with APR)	ContextNet v1 (default)	36.46/0.966	33.90/0.944
	ContextNet v2	36.50/0.966	33.86/0.944

Table 1: Ablations on different ContextNet designs. Detailed structures are shown in Fig. **??**.

Necessity of Alignment-aware Patch-level Routing. Instead of using predicted route map from APR, we also try to remove APR and directly adopt the reference route map for synthesis network during inference progress. The results in Tab. **??** show performance of VFI

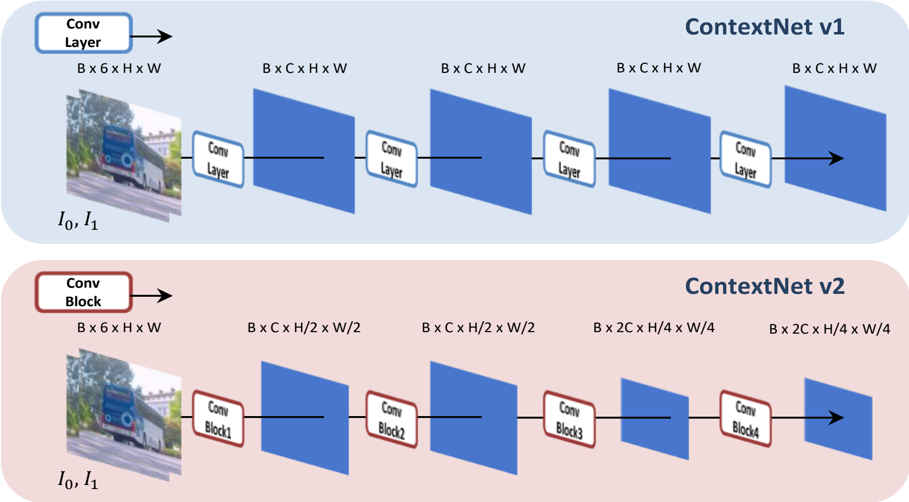

Figure 1: Two designs of context network.

Routing Module	batch size	Vimeo90k		X-2k		X-"4K"	
		PSNR/SSIM	keep ratio	PSNR/SSIM	keep ratio	PSNR/SSIM	keep ratio
Route Label Generation	1	35.78/0.979	49%	36.45/0.966	50%	33.88/0.944	50%
	4	35.80/0.979	50%	OOM	OOM	OOM	OOM
	8	35.81/0.979	50%	OOM	OOM	OOM	OOM
	16	35.81/0.979	50%	OOM	OOM	OOM	OOM
	64	35.82/0.980	50%	OOM	OOM	OOM	OOM
APR (ours)		35.82/0.980	48%	36.46/0.966	46%	33.90/0.944	51%

Table 2: Ablations on different version of routing module. Keep ratio means percentage of patch feeding into refine block (instead of copy) in synthesis network. OOM means out of memory.

guided by reference map is slightly lower than VFI with APR predicted on Vimeo90K and X-"4K". Moreover, the reference route map would be influenced by batch size, leading to unstable interpolation results. Therefore, training an Alignment-aware Patch-level Routing Module is essential for dynamic VFI.

Effects of Training Strategy. We adopt 3-stage training strategy and freeze optical flow network during the last two stages. The comparison results in Tab. **??** report that training all model components in one step slightly degrades performance (line 2). Meanwhile, freezing optical flow network brings improvement because it prevents estimated flow network from being disrupted by significant increased loss due to APR.

Effects of APR Loss Ratio. APR loss ratio controls the tolerance of distance between APR predicted route map with pre-defined reference route map. In Tab. **??**, a smaller APR loss ratio provides a loose constraint to APR module, encouraging more patches (more computation) to be convolved in synthesis network. When APR loss ratio = 0, APR-VFI would degrade to static VFI, because choosing all patches is the easiest method to improve performance. In our experiments, we choose 0.01, because it achieves the best trade-off on Xiph.

Performance on Vimeo90k. Vimeo90k is not our focus, but we provides our corresponding performance and FLOPs in Tab. **??**. Our dynamic APR-VFI surpasses dynamic VFI UGSP [**?**] by 0.1db. Our static VFI(w/o APR) achieves similar performance with FILM [**?**] and EBME-H [**?**]. However, we do not outperform IFRNet large [**?**], ABME [**?**] and Softsplat [**?**]. In the future, we will optimize model architecture to improve perfor-

curriculum training	freeze optical flow network	X-2k	X-4k
✓		36.38/0.965	33.81/0.943
	✓	36.44/0.966	33.88/0.944
✓	✓	36.46/0.966	33.90/0.944

Table 3: Ablations on training strategy

APR loss ratio	X-2k		X-4k	
	PSNR/SSIM	keep ratio	PSNR/SSIM	keep ratio
1	36.45/0.965	38.66%	33.85/0.943	40.63%
0.1	36.45/0.965	42.23%	33.85/0.944	46.44%
0.01	36.46/0.966	46.11%	33.90/0.944	50.63%
0.001	36.59/0.966	79.01%	33.95/0.945	83.33%

Table 4: Ablations on different APR loss ratios. Keep ratio means percentage of patch feeding into refine block (instead of copy) in synthesis network.

Model	Vimeo90k	GFLOPs
ABME [**?**]	36.18/0.981	161.7
CAIN [**?**]	34.65/0.973	167.1
SepConv [**?**]	33.79/0.970	109.9
SuperSloM [**?**]	34.35/0.957	156.7
AdaCoF [**?**]	34.47/0.973	44.6
DAIN [**?**]	34.71/0.976	702.1
FILM [**?**]	36.06/0.970	250.2
EBME-H [**?**]	36.06/0.980	55.9
SoftSplat [**?**]	36.10/0.980	114.2
IFRNet large [**?**]	36.20/0.981	101.1
Ours (w/o APR)	36.05/0.980	96.7
UGSP [**?**]	35.72/-	21.0
Ours (with APR)	35.82/0.980	63.9

Table 5: Model performance on Vimeo90K. "-" means corresponding data is unavailable. The FLOPs is measured with 256×448 resolution. Noted that Vimeo90k is not our focus, we focus on large motion dataset.

mance.

2 Visualization Result

Comparison of Model Results. We provide more visualization results on Xiph dataset in Fig. **??**. It shows that our method provides the best visual effects.

Route Map. The Fig. **??** shows the predicted route map from APR. Large motion regions would pass through more layers.

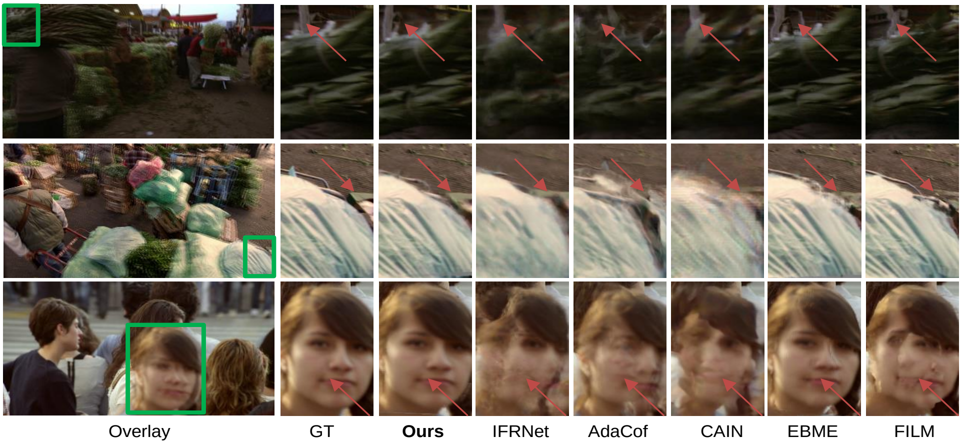

Figure 2: Comparison results on X-2k.

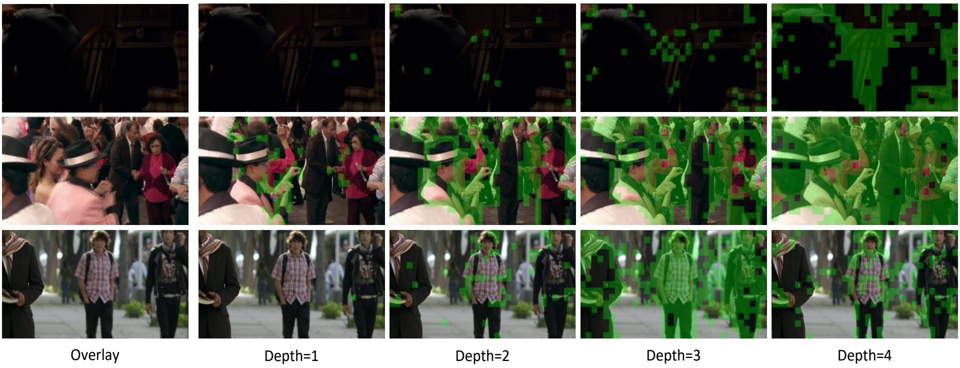

Figure 3: Visualization of route map predicted by APR. The green masks at different depths represent the patches passing through corresponding refine block.

References

[] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3703–3712, 2019.

[] Ri Cheng, Xuhao Jiang, Ruian He, Shili Zhou, Weimin Tan, and Bo Yan. Uncertainty-guided spatial pruning architecture for efficient frame interpolation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1975–1986, 2023.

[] Myungsub Choi, Heewon Kim, Bohyung Han, Ning Xu, and Kyoung Mu Lee. Channel attention is all you need for video frame interpolation. In *AAAI*, 2020.

[] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan 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*, pages 9000–9008, 2018.

[] Xin Jin, Longhai Wu, Guotao Shen, Youxin Chen, Jie Chen, Jayoon Koo, and Cheul-hee Hahm. Enhanced bi-directional motion estimation for video frame interpolation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023.

[] Lingtong Kong, Boyuan Jiang, Donghao Luo, Wenqing Chu, Xiaoming Huang, Ying Tai, Chengjie Wang, and Jie Yang. Ifrnet: Intermediate feature refine network for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. Adacof: Adaptive collaboration of flows for video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5316–5325, 2020.

[] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[] Junheum Park, Chul Lee, and Chang-Su Kim. Asymmetric bilateral motion estimation for video frame interpolation. In *International Conference on Computer Vision*, 2021.

[] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *European Conference on Computer Vision*, pages 250–266. Springer, 2022.