Deep Video Matting via Spatio-Temporal Alignment and Aggregation

Supplementary Material

1. ST-FAM

In temporal feature alignment module (TFA), we apply deformable convolution layer to align features. Figure 1 shows the implementation. For simplicity, we denote the encoded features extracted from the resblocks at timestamp t as $\mathcal{F}_t$. To reduce computational overhead, before sending the encoded feature to TFA, we apply a 1×1 convolution layer to reduce their channel to 64. Then we perform two 3×3 convolution layers on the concatenation of $\mathcal{F}_i$ and $\mathcal{F}_{i+1}$ ($i \in [-n, n]$). The learned features are split into offset and mask, which are sent to a deformable convolution [3] layer. After we have obtained $2n + 1$ aligned features for all frames, a temporal feature fusion module (TFF) with channel- and spatial- attention is applied on these aligned features for aggregation. Figure 2 shows the structures of our channel- and spatial- attention used in TFF. n is set to 2 in our experiments.

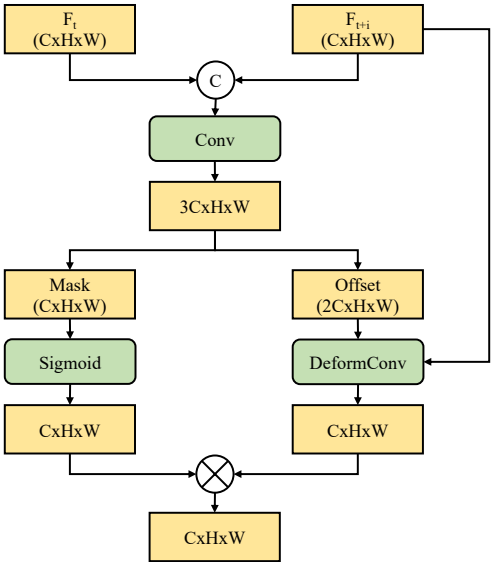

Figure 1. Architecture of deformable convolution in TFA. c is 64.

2. Experiments

2.1. Temporal Aggregation

In this ablation study, we investigate the effect of temporal aggregation by varying the number n of neighboring frames. Note that the actual number of input frames is $2n + 1$. To aggregate more neighboring frames within limited memory and shorten the training time, we use ResNet-34 [1] as our encoder, and train all models for 60 epochs with a batch size of 1. Table 1 shows the results. The performance is promoted as we increase the number of neighboring frames. By integrating more neighboring frames, our model can have a better view of objects' motion to generate more precise predictions. With the number continues to increase, the performance of our model holds steady as the model has learned sufficient short-term temporal information from neighboring frames.

n	1	2	3	4	5	6
SAD	48.72	46.98	46.29	46.30	46.34	46.32
dtSSD	19.24	18.59	18.17	18.10	18.09	18.10

Table 1. Experimental results of temporal aggregation.

2.2. Temporal Fusion Network

In this ablation study, we compare different temporal fusion networks with our ST-FAM, including naive-fusion, cross-attention-fusion and flow-fusion. Their structures are described as following.

Naive-fusion. Given $\mathcal{F}_t$ and $\mathcal{F}_{t+i}$ ($i \in [-n, n]$), naive-fusion takes their concatenation as input, and applies two 3×3 convolutional layers to generate the aggregated temporal feature.

Cross-attention-fusion. Similar to the cross-attention layer used in our trimap propagation network, in this model, we apply a cross-attention layer to enhance $\mathcal{F}_t$ by $\mathcal{F}_{t+i}$ ($i \in [-n, n]$). Figure 3 shows the implementation, where c is also reduced to 64 and n is set to 2 in our experiments. The $2n+1$ enhanced features are then concatenated and utilized in the decoder. Experimental results demonstrate the effectiveness of cross-attention in temporal aggregation at the expense of large computational overhead which limits extension to more neighboring frames.

Flow-fusion. Different from the aforementioned fusion methods that encode temporal information by integrating multi-frame features, flow-fusion incorporates temporal information by the concatenation with optical flow. Specif-

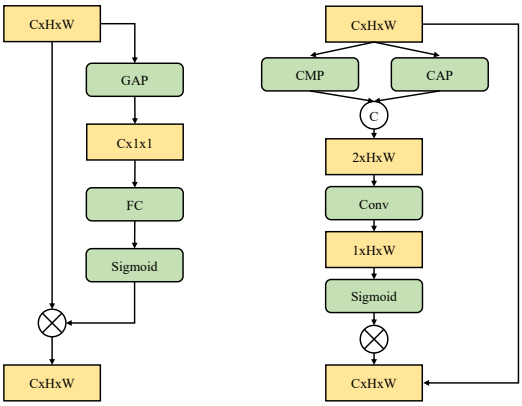

Figure 2. Left: channel attention. Right: spatial attention. GAP: global average pooling. CMP: channel maximum pooling. CAP: channel average pooling. c is $(2n+1) \times 64$ and n is the number of neighboring frames used in the paper.

Diff	Grad	KL	TC	SAD	MSE	dtSSD
✓	-	-	-	42.82	0.015	16.42
✓	✓	-	-	41.96	0.015	16.24
✓	✓	✓	-	41.47	0.014	15.85
✓	✓	✓	✓	40.91	0.014	15.11

Table 2. Experimental results of our video matting framework with different losses. "Diff", "Grad", "KL", "TC" respectively represents difference loss, gradient loss, KL-divergence loss and temporal coherence loss.

ically, this method applies an off-the-shelf flow estimation network PWC-Net [2] on two RGB frames at time t and $t+i$ ($i \in [-n, n]$), and generates corresponding optical flows. Afterward, all flows are concatenated with the decoded features to offer motion information of foreground objects and background scenes. However, the promotion is limited due to the ambiguous flow estimation within regions comprised of many semi-transparent/transparent pixels.

2.3. Losses

In the training stage, compared to former methods, we apply multiple losses to optimize our model. We conduct several experiments to illustrate their effectiveness. We take the model with different losses, including alpha prediction loss and composition loss as the baseline, and progressively add gradient loss, KL-divergence loss and temporal coherence loss. Table 2 tabulates the experimental results. The gradient loss can guide our model to pay attention to difficult structures or patterns leading to a 0.86 SAD improvement. Moreover, the KL-divergence loss minimizes the discrepancy between the distribution of predictions and targets, which further improves the performance by 0.49.

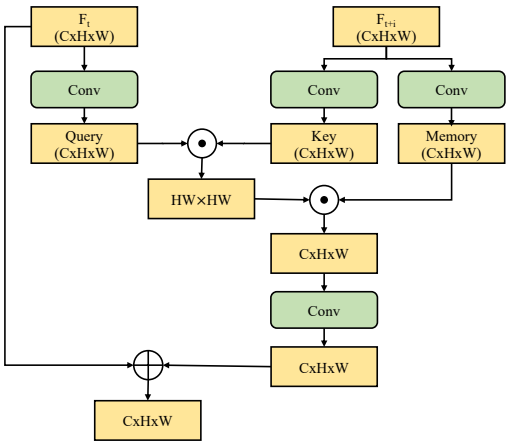

Figure 3. Architecture of Cross-attention-fusion. c is 64.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[2] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *CVPR*, 2018. 2

[3] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets V2: more deformable, better results. In *CVPR*. 1