Appendix for AccFlow: Backward Accumulation for Long-Range Optical Flow

A. Implementation Details

In Section 3.3 and Figure 4 of the manuscript, we introduced the Motion Encoder, Motion Decoder, adaptive blending, and $getOcc$ modules used in our proposed AccFlow framework. Due to the page limitation, we provide implementation details of these modules in this Appendix.

As shown in Figure A, the Motion Encoder and Motion Decoder used in AccPlus are implemented where 'Conv $k \times k$ (c)' represents a convolutional layer with kernel size $k \times k$ and output channel c, and 'ReLU' stands for the activation function. In the Motion Encoder, the input optical flow of size $w \times h \times 2$ first undergoes bilinear sampling to spatially downscale the flow in the 'DownsampleX4' layer, followed by a set of convolutional layers that produce the motion feature of size $w/8 \times h/8 \times 128$. In the Motion Decoder, the optical flow is inferred in parallel from the input motion feature of size $w/8 \times h/8 \times 128$, following the approach in RAFT. The left branch in Figure A-(b) produces a optical flow of size $w/8 \times h/8 \times 2$, while the right branch produces a weighting mask of size $w/8 \times h/8 \times 144$. The convex upsampler is used to upscale the optical flow by $8\times$ using the adaptive weighting mask (with kernel size 3×3) from the left branch.

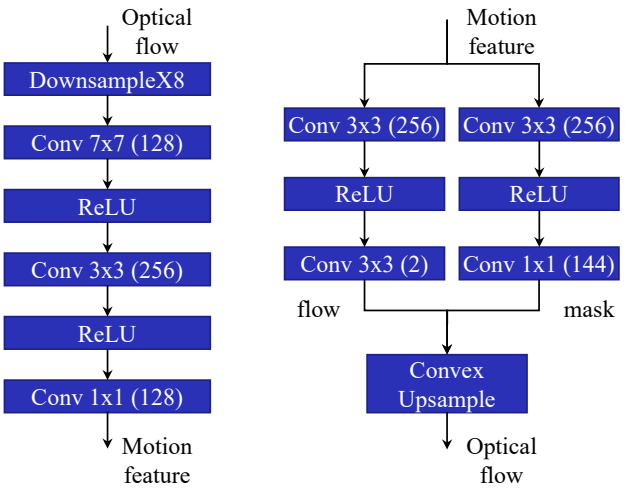
The detailed implementation of adaptive blending module is presented in Figure B. The frames $\mathbf{I}_N$ and $\mathbf{I}_{t-1}$ are encoded to context features $\mathbf{C}_N$ and $\mathbf{C}_{t-1}$, which have a $1/4$ spatial size of the frame. The initialized long-range motion feature $f^{ini}_{t-1,N}$ is mapped to pixel offsets for deformable convolution, which aligns the context feature $\mathbf{C}_N$ to $\widetilde{\mathbf{C}}^{t-1}_N$. Afterwards, the L1 difference $|\mathbf{C}_N - \widetilde{\mathbf{C}}^{t-1}_N|$ generates an attention mask $m \in \mathbb{R}^{h/8 \times w/8 \times 1}$, which adaptive determines how the prior feature $f^{ini}_{t-1,N}$ rectifies the accumulated error in $f_{t-1,N}$.

In our current AccFlow framework, we implement the $getOcc$ function by using a threshold to generate the occlusion mask. Specifically, given the optical flow $\mathbf{F}_{i,k}$, we backward warp frame $\mathbf{I}_k$ to generate $\widetilde{\mathbf{I}}^i_k$, Afterwards, the L1 difference between $\mathbf{I}_i$ and $\widetilde{\mathbf{I}}^i_k$ is obtained as follows:

$$e = |\mathbf{I}_i - \widetilde{\mathbf{I}}^i_k|, \tag{1}$$

where $e \in \mathbb{R}^{w \times h \times 3}$. Subsequently, we obtain the average error mask along e's channel dimension and mark the regions where the mean value is greater than 125 as occluded. Although the aforementioned $getOcc$ implementation may not accurately classify occluded and visible regions, we adopt this approach since it is very efficient and can already provide adequate occlusion estimates for the subsequent occlusion solvers. Finally, we spatially downscale the occlusion mask using nearest-neighbor sampling by $1/8$ times.

In Section 4.5, we conduct an ablation study comparing the forward version of AccFlow, denoted as AccFlow (F.), with its default backward version. Figure C illustrates the structure of AccFlow (F.), in which we keep the network structure and parameter number the same as backward accumulation but change the input order.



(a) Motion Encoder (b) Motion Decoder

Figure A: Structure of the Motion Encoder and Decoder in Section 3.3 (Figure 4) of the manuscript.

B. Supplementary Visual Comparison

In this section, we present additional visual quality comparisons to demonstrate the effectiveness of our proposed method. Specifically, we compare the long-range flow estimated from the HS-Sintel dataset in Figure D and E, and the estimation from our CVO testing sets in Figure F and Figure G. To ensure fair comparisons, we fine-tune

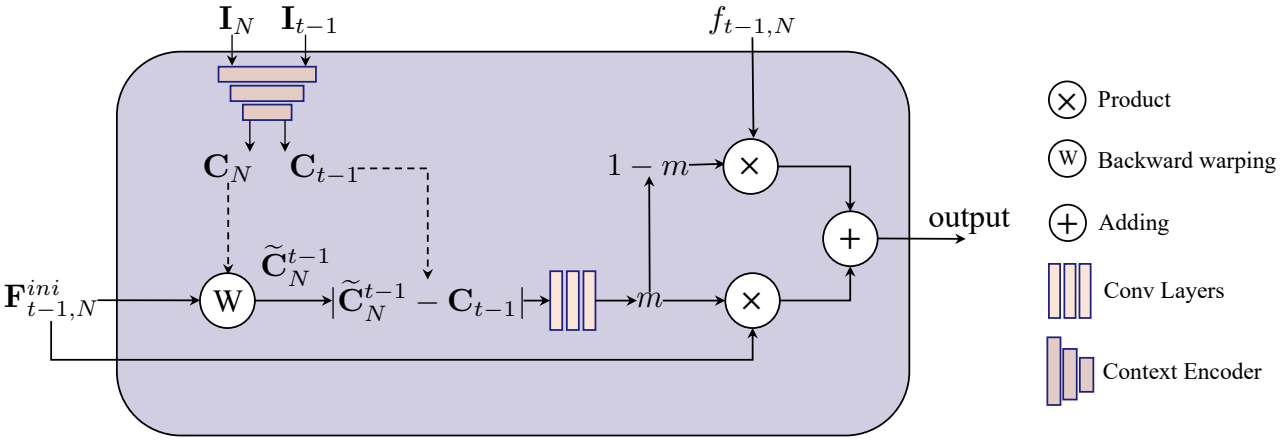
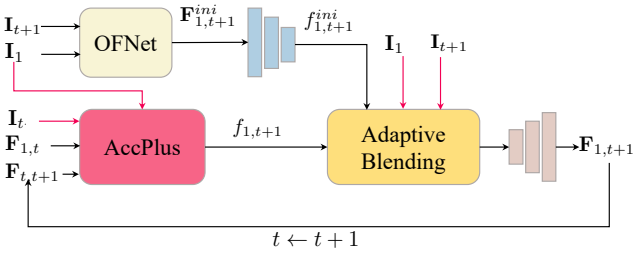
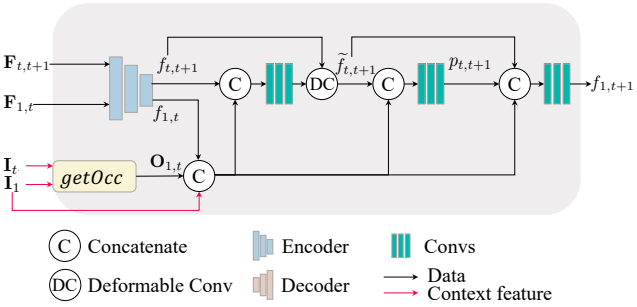


Figure B: Structure of the Adaptive Blending in Figure 4 (Section 3.2) of the manuscript.



(a) AccFlow (F.) Framework.



(b) AccPlus (F.) Module.

Figure C: Illustration of the forward version AccFlow framework. (a) The AccFlow (F.) framework. Time t increases from 2 to $N-1$ to obtain the long-range flow $\mathbf{F}_{1,N}$. (b) The AccPlus (F.) module implements the forward accumulation in feature domain. Compared with Figure 1 of the manuscript, we only change the input order.

the pretrained RAFT and GMA models on our CVO training set, denoted as RAFT* and GMA*, respectively. We also present the results from warm-start (denoted as '-*w*') and AccFlow (*i.e.*, Acc+GMA*). On the HS-Sintel testing dataset, given two distant frames and their ground-truth long-range optical flow, we present the flow estimation, the corresponding error maps (with darker areas indicating less error), and the zoom-in error maps in Figure D and E. Similarly, we present the visual comparison on our CVO *Clean* and *Final* sets in Figure F and G, respectively. Our results demonstrate that warm-start improves the performance compared to the direct estimation, while our proposed AccFlow produces more accurate results and outperforms other methods by a large margin, especially in the cases with complex occlusion and small objects with large motion.

C. Limitation

While our proposed CVO dataset contains challenging scenes with large motion and occlusion, its appearance and motion patterns may differ from real-world scenes since it is a synthetic dataset. This limitation may reduce the generalization of our learned model in real-world scenarios. Furthermore, our proposed AccFlow method requires datasets that include ground-truth long-range flows for training, which limits the direct applicability of our model to many popular optical flow training datasets that only provide ground-truth local flows.

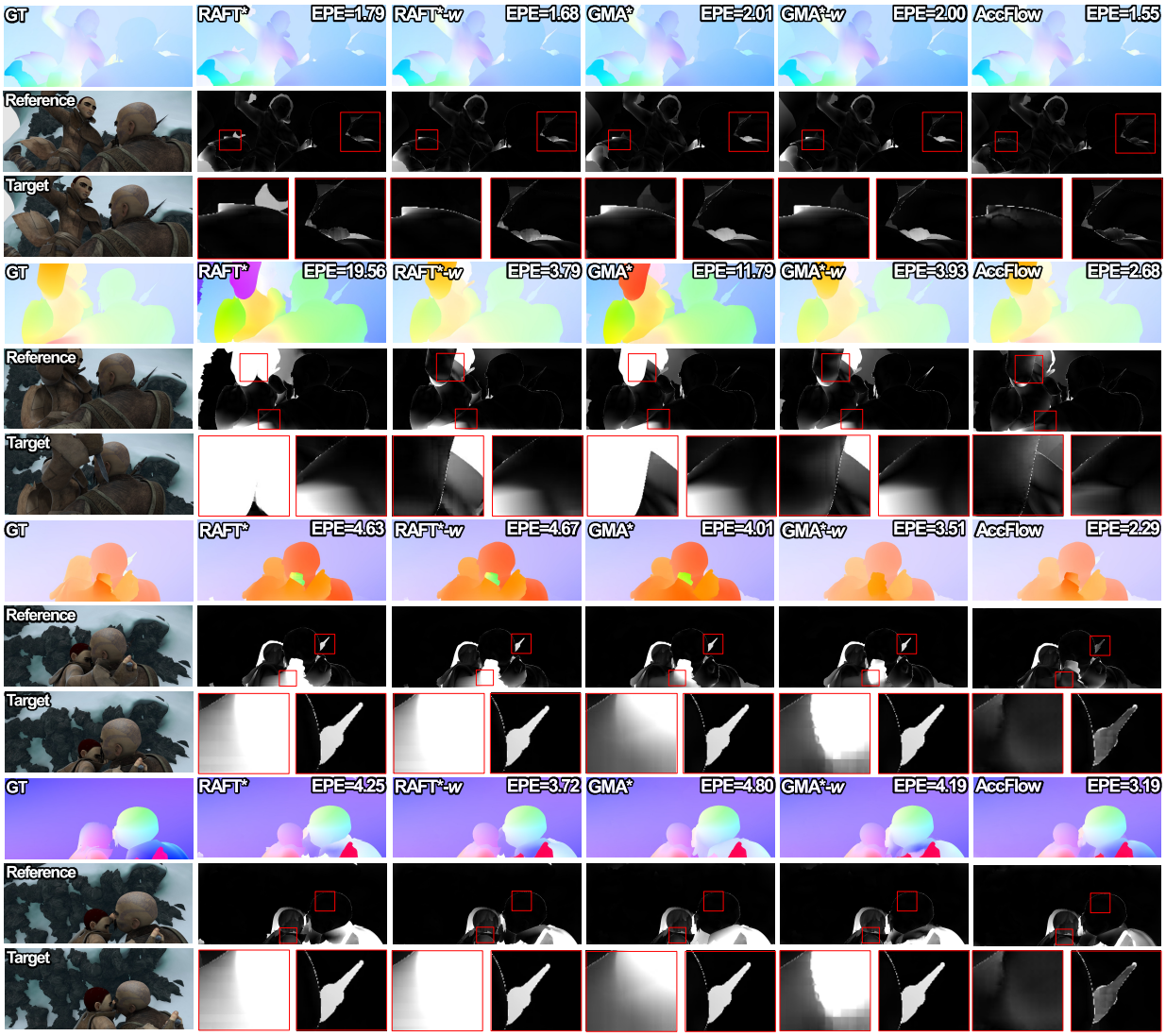


Figure D: Visual quality comparisons on HS-Sintel dataset.

Figure E: Visual quality comparisons on HS-Sintel dataset.

Reference | Target | GT | RAFT* | RAFT*-*w* | GMA* | GMA*-*w* | AccFlow

Figure F: Visual quality comparisons on CVO (*Clean*) testing set.

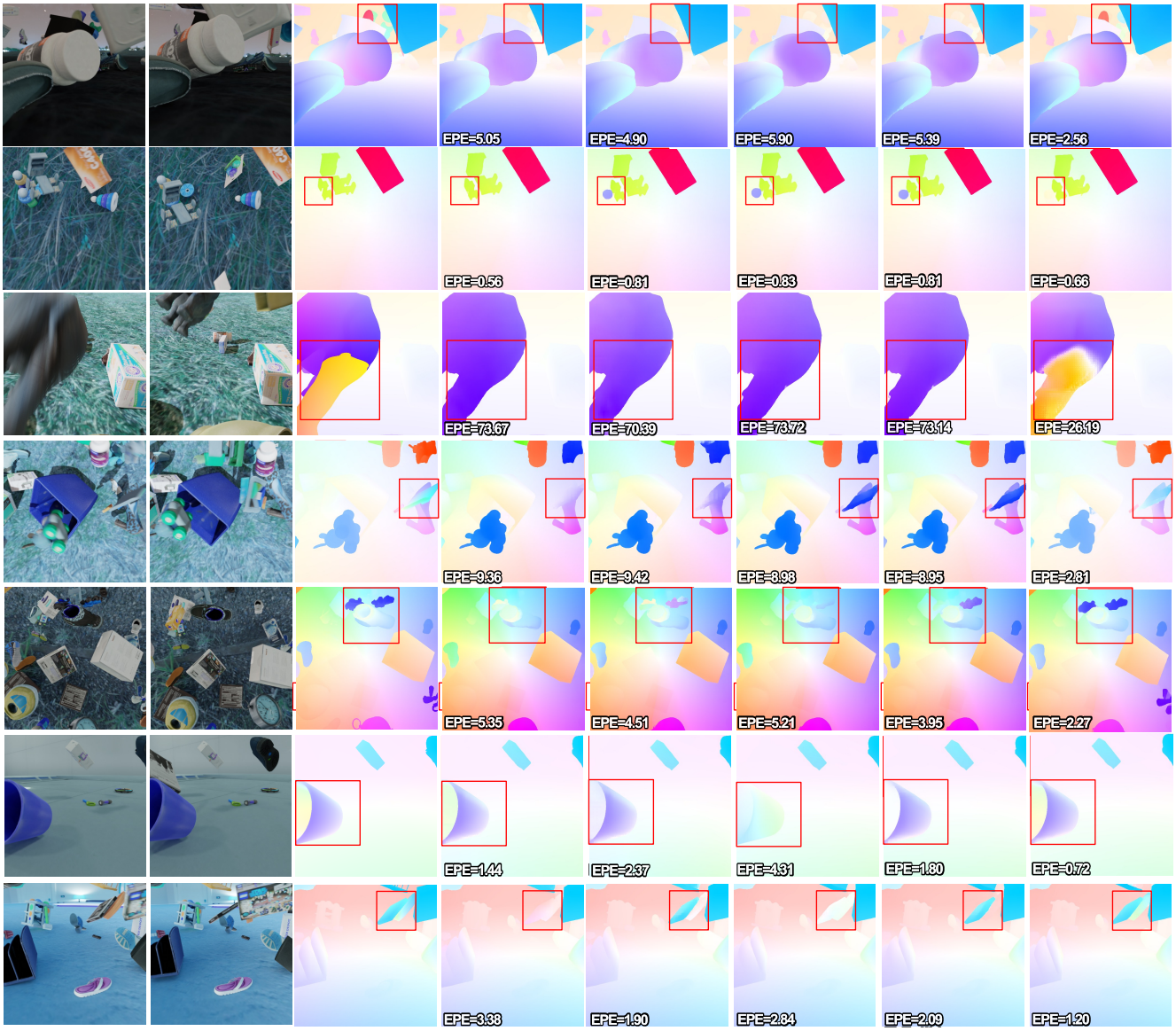


Figure G: Visual quality comparisons on CVO (*Final*) testing set.