A. Implementation Details

General architectural configuration We adapt our architectures from Zhu et al. [11] and Johnson et al. [5]. For all experiments described in the main paper, we use 5 blocks for the encoder and 5 blocks for the decoder. Below, we follow the naming convention used in their Github repositories to describe our general architectural configuration.

Let \(c_{MsN-K}\) denote a \(M \times M\) Conv-BN-Activation layer with stride \(N\) and \(K\) filters. We use Inplace-ABN [10] to reduce the memory consumption. Further, let us define an encoder basic block \(e_{M-K}\) by cascading \(c_{Ms1-K}\) with another downsample convolution block \(c_{Ms2-K}\) where ReLU is used. The basic decoder block \(d_{M-K}\) consists of a nearest-neighbor upsample layer followed by two \(c_{Ms1-K}\) layers in which activation layers are chosen as LeakyReLUs of slope 0.2.

Flow Predictor Our flow predictor \(F\) could be defined as:

\[
e_{7-64}, e_{5-128}, e_{5-256}, e_{3-512}, e_{3-512}, d_{3-512}, d_{3-512}, d_{3-256}, d_{3-128}, d_{3-2},
\]

where the last output layer has no activation, i.e., the flow prediction network regresses unconstrained displacement values for each coordinates. Raised by [4], we also empirically confirmed large kernel sizes, in first several layers, help the training to converge.

Occlusion Inpainter Our occlusion inpainter uses the same architectural parameters as in the flow predictor. The only differences here are that: (1) we replace the normal convolution operators with partial convolution operators in all \(e_{M-K}\)’s and fusion convolution operators in all \(d_{M-K}\)’s; (2) we replace \(d_{3-2}\) with \(d_{3-3}\), where Tanh activation is used to bound the output value between \(-1\) and 1.

B. Training Details

Here we specify more training details to supplement what we have described in the main paper. To train the flow predictor, we start from the learning rate at \(10^{-4}\) and decay it by \(1/10\) at the half of the training epochs, then repeat it again at the 3/4 of the training epochs. The occlusion inpainter is trained from \(10^{-3}\) and scheduled with the same decay strategy. We train our flow predictor, and occlusion inpainter for 200 epochs, and 800 epochs on CalTech Pedestrian dataset [2]. For KITTI Flow dataset [9], they are trained for 500 and 1000 epochs, respectively.

C. Supplementary Results

In this section, we include more results to supplement our main paper. We include more qualitative results for both Next-Frame Prediction and Multi-Frame Prediction. We also include the quantitative results for SSIM evaluations on Multi-Frame Prediction tasks on KITTI Flow dataset. To better assess our prediction results please refer to our website.

Next-Frame Prediction More qualitative results are shown in Figure S1 and S2, respectively. The experiment settings are consistent with the setups we established in the main paper.

Multi-Frame Prediction We here show comparison results for Multi-Frame Predictions in Figure S3. The experiment setting is 4-in 8-out prediction task. Compared to prior work [3] whose flow prediction degrades dramatically after a few time steps, our model can remain high fidelity even at the last several frames. We here also include the quantitative results for SSIM evaluations in S4.

D. Supplementary Ablations

In Table S1, we add one more ablation study to resolve the concern about the auxiliary losses we applied to our model. We build three groups of comparison experiment by removing perceptual and style losses, segmentation loss, or replacing our fusion decoder with normal partial convolutions as in [6]. All generators are trained using the same oracle model used in motion ablation studies. Removing perceptual and style losses does not hurt the performance of PSNR, but leads to large degeneration in structural and perceptual metrics. On the other hand, removing segmentation loss and our fusion decoding blocks results in performance drops in all metrics.

1All ReLU units are approximated by LeakyReLUs of slope 0.01 to be compatible with Inplace-ABN [10]

2https://sites.google.com/view/fgvp
Figure S1: More qualitative comparisons for 10-in 1-out Next-Frame Prediction on CalTech Pedestrian dataset.

Figure S2: More qualitative comparisons for 4-in 1-out Next-Frame Prediction on KITTI Flow dataset.
Figure S3: Qualitative comparisons for 4-in 8-out Multi-Frame Prediction on KITTI Flow dataset.

Figure S4: SSIM↑ quantitative results for 4-in 8-out Multi-Frame Prediction on KITTI Flow dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR↑</th>
<th>SSIM↑</th>
<th>LPIPS↓ ($\times 10^{-2}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o p+s</td>
<td>23.3</td>
<td>0.748</td>
<td>17.9</td>
</tr>
<tr>
<td>w/o seg</td>
<td>22.6</td>
<td>0.766</td>
<td>10.2</td>
</tr>
<tr>
<td>w/o fus</td>
<td>22.42</td>
<td>0.760</td>
<td>10.8</td>
</tr>
<tr>
<td>all</td>
<td>23.3</td>
<td>0.786</td>
<td>9.9</td>
</tr>
</tbody>
</table>

Table S1: Supplementary ablation study on auxiliary losses in our model.
References


