Video Frame Interpolation via Adaptive Convolution

Simon Niklaus*, Long Mai*, Feng Liu

Portland State UNIVERSITY

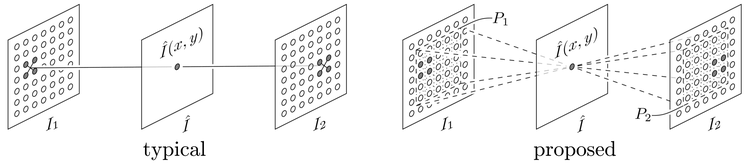
CVPR July 21-26 HONOLULU 2017

Overview

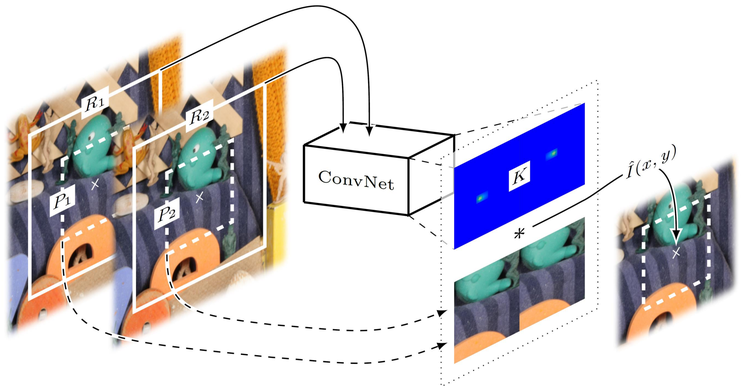
Video frame interpolation typically involves two steps: motion estimation and pixel synthesis. Such a two-step approach heavily depends on the quality of motion estimation. This paper presents a robust video frame interpolation method that combines these two steps into a single process.

Specifically, our method considers pixel synthesis for the interpolated frame as local convolution over two input frames. The convolution kernel captures both the local motion between the input frames and the coefficients for pixel synthesis.

Flow-based vs convolution-based interpolation

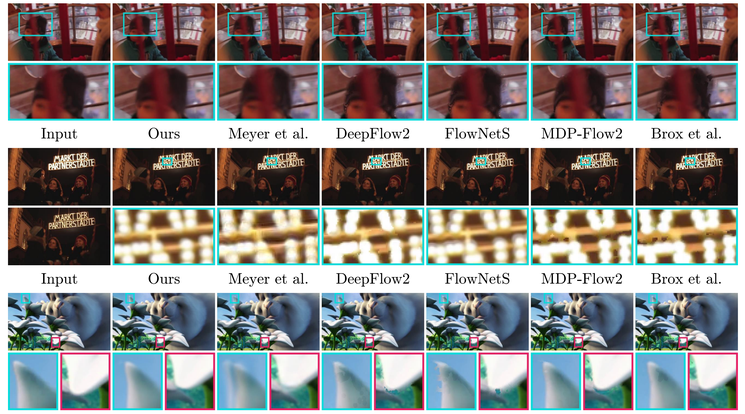


(1) The combination of motion estimation and pixel synthesis into a single step provides a more robust solution. **(2)** The convolution kernel provides flexibility to account for and address difficult cases like occlusion. **(3)** If properly estimated, the convolution formulation can seamlessly integrate advanced resampling techniques like edge-aware filtering.



$$\hat{I}(x,y) = K_1(x,y) * P_1(x,y) + K_2(x,y) * P_2(x,y)$$

Comparison



Method

We utilize a fully convolutional neural network to estimate the convolution kernels for each individual output pixel.

$$\text{color loss: } \sum_i \|[P_{i,1}\ P_{i,2}] * K_i - \tilde{C}_i\|_1$$

$$\text{gradient loss: } \sum_i \sum_k \|[G^k_{i,1}\ G^k_{i,2}] * K_i - \tilde{G}^k_i\|_1$$

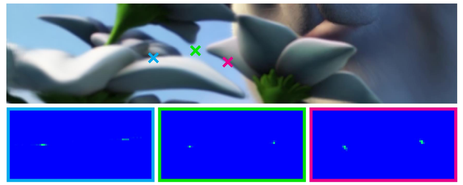
By using a color loss together with a gradient loss, we achieved sharper interpolation results.

Training data can be obtained by collecting frames from videos, grouping them into triplets and using the middle frame as ground truth.

Evaluation

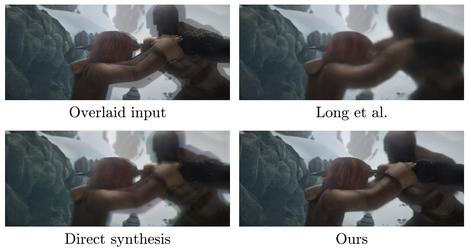
	Mequ.	Schef.	Urban	Teddy	Backy.	Baske.	Dumpt.	Everg.
Ours	3.57	4.34	5.00	6.91	**10.2**	**5.33**	**7.30**	**6.94**
DeepFlow2	2.99	3.88	**3.62**	5.38	11.0	5.83	7.60	7.82
FlowNetS	3.07	4.57	4.01	5.55	11.3	5.99	8.63	7.70
MDP-Flow2	**2.89**	**3.47**	3.66	**5.20**	**10.2**	6.13	7.36	7.75
Brox *et al.*	3.08	3.83	3.93	5.32	10.6	6.60	8.61	7.43

average interpolation error in the Middlebury benchmark



estimated kernels that are spatially adaptive and edge-aware

Ours vs direct synthesis



We experimented with a baseline by modifying our network to directly synthesize pixels. We found the results of our baseline and another direct method [Long et al.] to be blurry.

Acknowledgements

The shown examples are based on footage from the Blender Foundation and the city of Nuremberg. The presented work was supported by NSF IIS-1321119.

http://graphics.cs.pdx.edu/project/adaconv