ComputerVisionLab
Seoul National University
http://cv.snu.ac.kr

Channel Attention Is All You Need for Video Frame Interpolation

Myungsub Choi[1] Heewon Kim[1] Bohyung Han[1] Ning Xu[2] Kyoung Mu Lee[1]
Dept of ECE, ASRI, Seoul National University [1] Amazon Go [2]

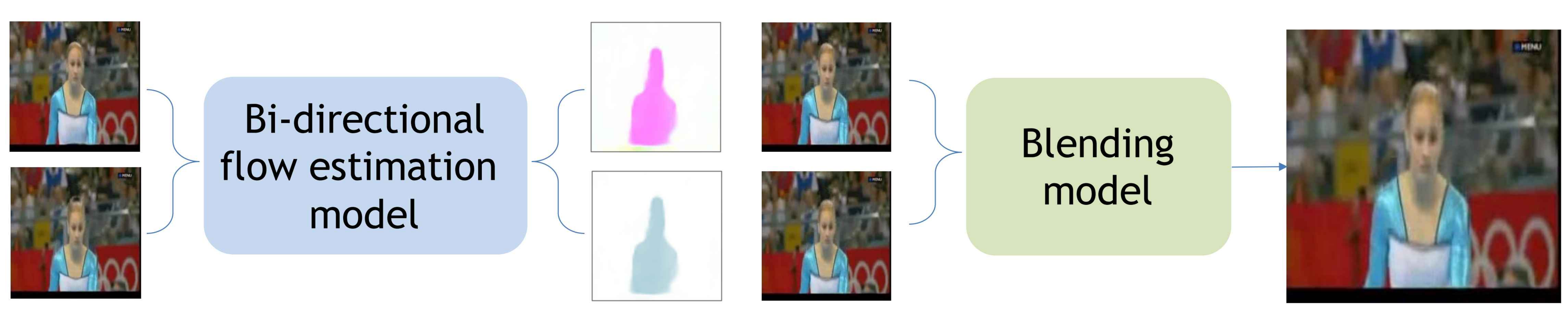
INTRODUCTION

Video frame interpolation

- Goal: Interpolate intermediate frames given two consecutive video frames
- Use cases: frame-rate up-conversion, slow motion effect, etc.

Interpolating process of previous methods [1, 3, 4]

1. Motion estimation
2. Align (warp) 2 input frames to middle for interpolation
3. Blend & Synthesize occluded regions



Motivation

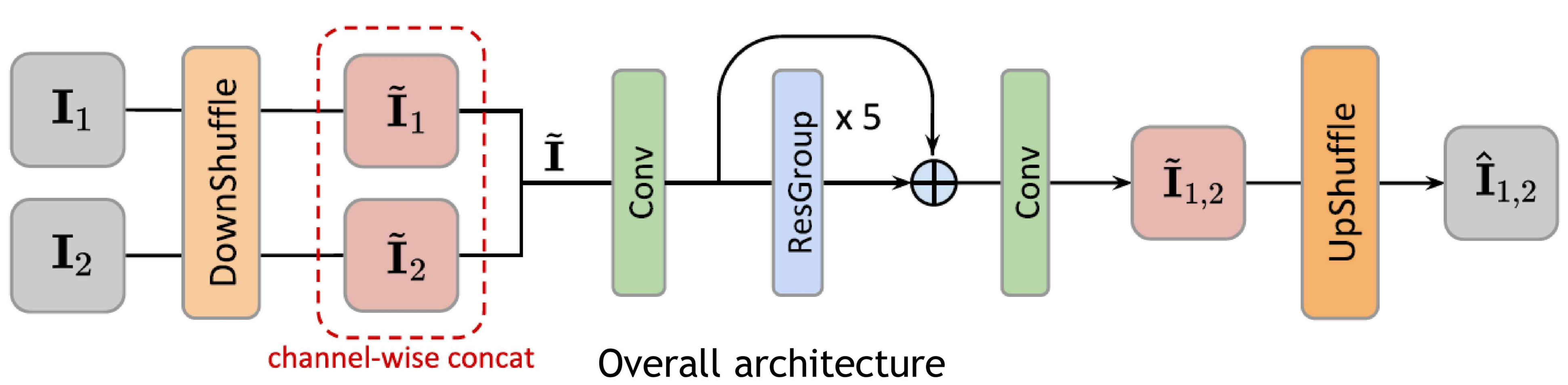
Input Overlay | PWC-Net | SepConv [2] | TOFlow [3] | CAIN (Ours)

Large motion / heavy occlusion at motion boundaries lead to severe ghost artifacts if flow estimation fails

→ If errors from wrong motion estimation propagate, **why not estimate pixels directly?**

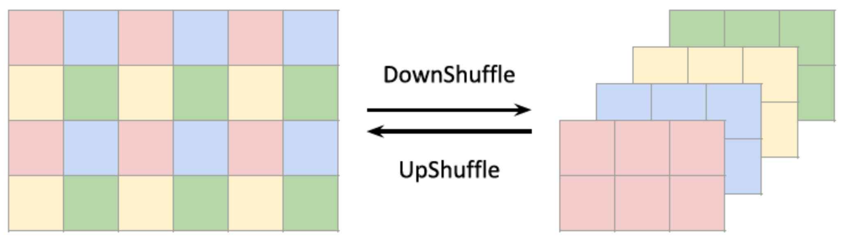
MODEL

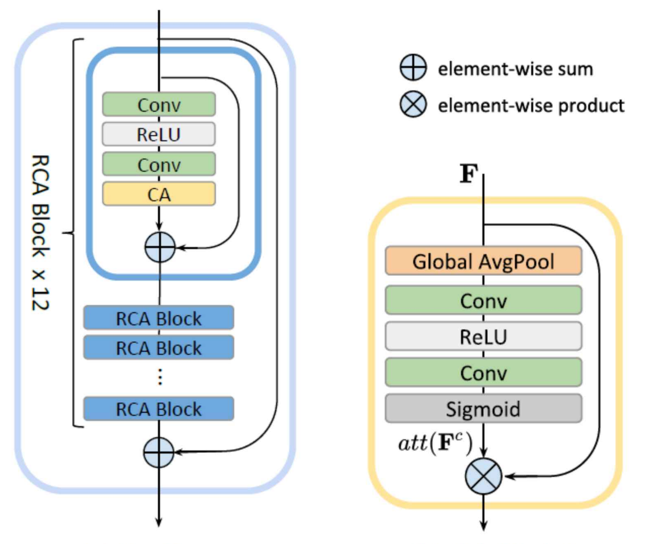
Our design philosophy : Distribute spatial motion to channel dimension
Then, use attention model to handle complex motion



Overall architecture

For spatial ⟷ channel, we use PixelShuffle →



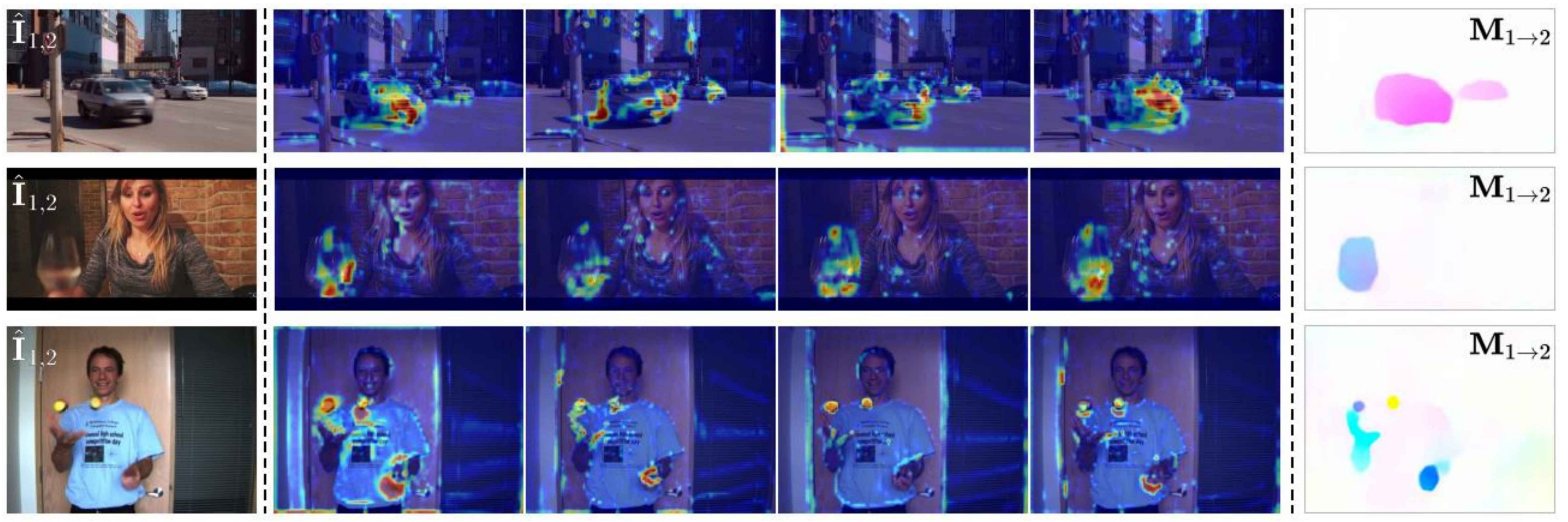


(a) ResGroup (b) CA Module

← For main interpolation model, we use stacked **residual blocks with channel attention**.

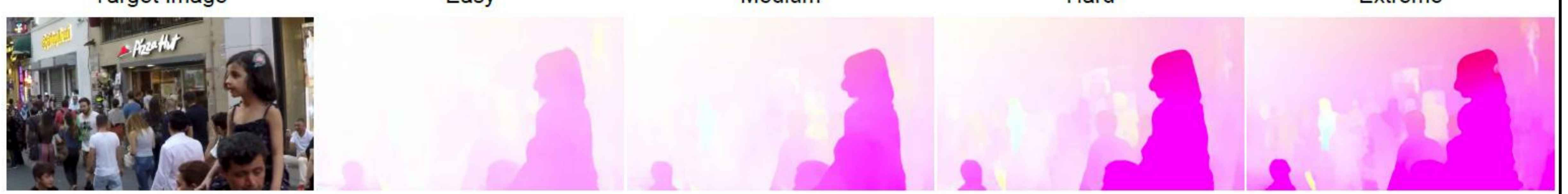
Since our model is composed of only **basic operations** (e.g. convolutions, skip connections, shuffling), it is very **simple & efficient**.

Visualization of feature activations for high (channel) attention



EXPERIMENTS

SNU-FILM (Frame Interpolation with Large Motion) evaluation benchmark



- Composed of 240fps video frames from GOPRO [5] test dataset + videos from YouTube
- 4 different evaluation settings *w.r.t.* motion magnitude

Quantitative results

Model	FILM(Easy)	FILM(Medium)	FILM(Hard)	FILM(Extreme)	Vimeo90K	UCF101	Middlebury
PWC-Net	36.42 / 0.983	33.09 / 0.960	27.72 / 0.888	23.81 / 0.806	31.36 / 0.939	33.60 / 0.963	2.24
DVF [1]	25.10 / 0.848	23.31 / 0.809	21.68 / 0.768	19.86 / 0.720	31.54 / 0.946	34.12 / 0.963	4.04
SepConv-L_1 [2]	39.68 / **0.990**	35.07 / 0.976	29.39 / 0.926	24.32 / 0.845	33.79 / 0.970	**34.95** / 0.968	**2.05**
TOFlow+Mask [3]	39.08 / 0.989	34.39 / 0.974	28.44 / 0.918	23.39 / 0.831	33.73 / 0.968	34.58 / 0.967	2.15
SuperSloMo [4]	†37.28 / 0.986	†33.80 / 0.973	†28.98 / 0.925	†24.15 / 0.845	†33.15 / 0.966	34.75 / 0.967	†2.28
CAIN (w/o CA)	39.59 / **0.990**	35.34 / 0.976	29.56 / 0.926	24.48 / 0.846	34.25 / 0.970	34.75 / 0.968	2.36
CAIN (Ours)	**39.78 / 0.990**	**35.49 / 0.977**	**29.86 / 0.929**	**24.69 / 0.850**	**34.65 / 0.973**	34.91 / **0.969**	2.28

†: results from publicly available implementation of (Paliwal 2018).

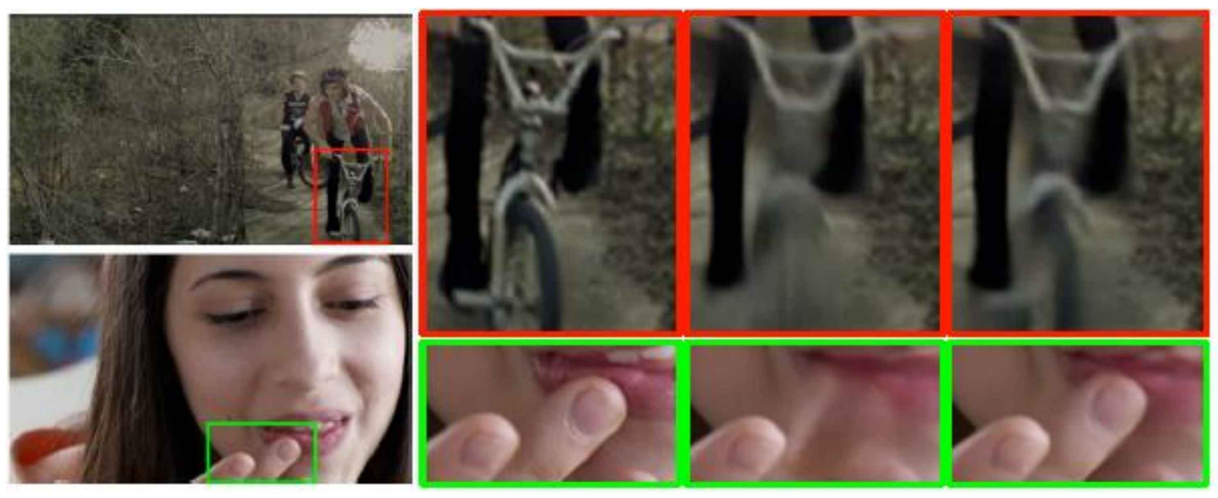
Visual results

Input overlay | DVF [1] | SepConv-L_1 [2] | SepConv-L_f [2] | TOFlow [3] | Ours | GT

Ablation study

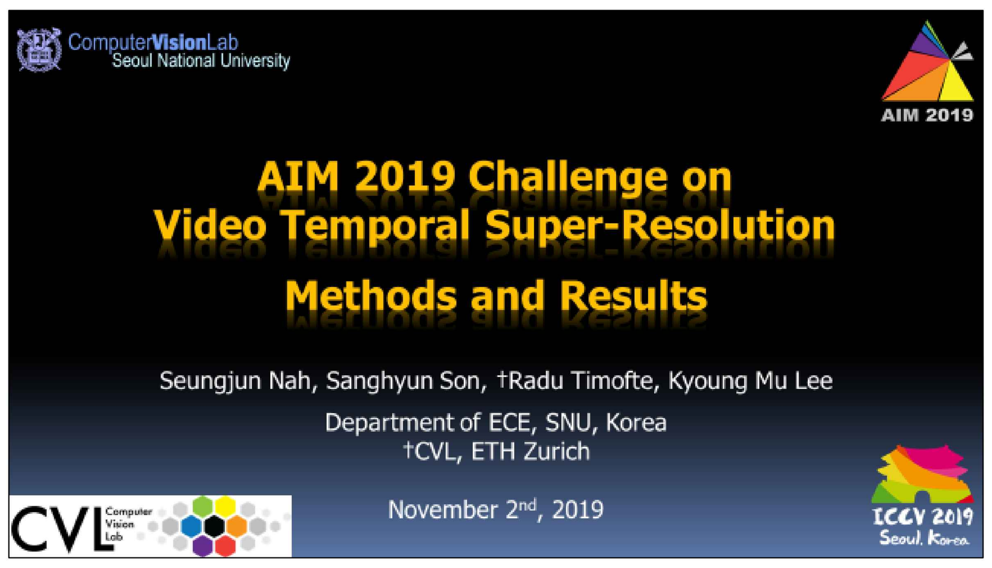
Channel attention helps to

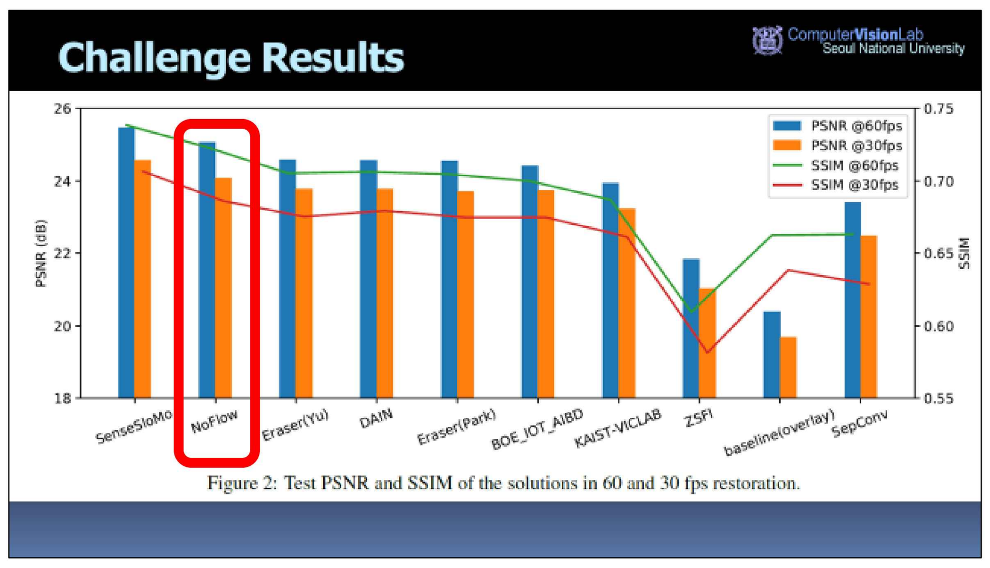
- Find the correct position
- Reduce blurry artifacts
- Achieve better performance



GT | w/o CA | w/ CA

ICCV 2019 – AIM 2019 Challenge





Won 2nd place as Team "NoFlow"

Code & Data available at: myungsub.github.io/CAIN/

REFERENCES

[1] Liu et al., Video frame synthesis using deep voxel flow. In ICCV 2017.
[2] Niklaus et al., Video frame interpolation via adaptive separable convolution. In ICCV 2017.
[3] Xue et al., Video enhancement with task-oriented flow. In CVPR 2018.
[4] Jiang et al., Super SloMo: High Quality Estimation of Multiple Intermediate Frames for Video Interpolation, In CVPR 2018.
[5] Nah et al., Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring, In CVPR 2017.