# Unsupervised Flow-Aligned Sequence-to-Sequence Learning for Video Restoration

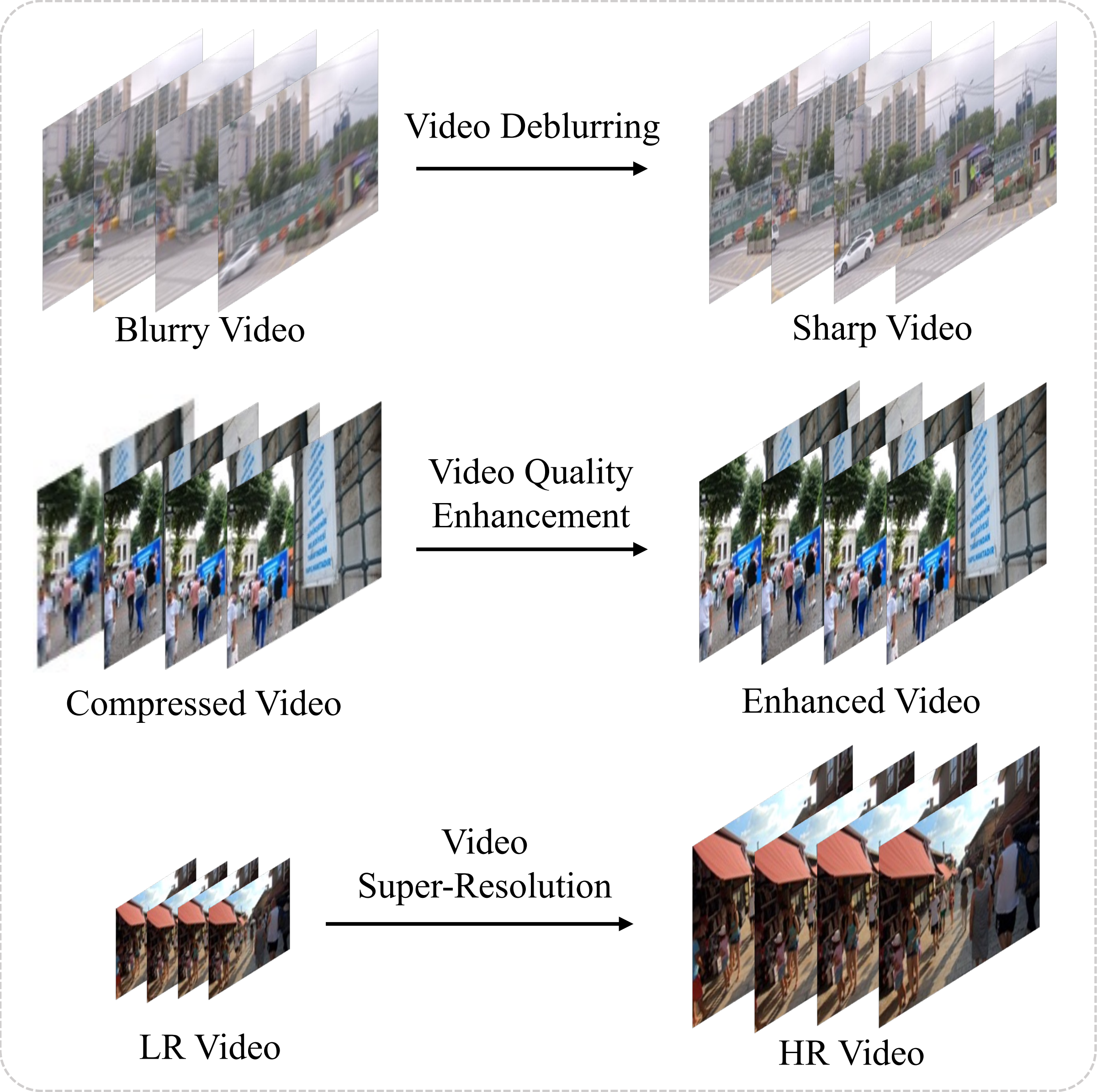
ICML 2022

Jing Lin[*1], Xiaowan Hu[*1], Yuanhao Cai[1], Haoqian Wang[†1]
Youliang Yan[2], Xueyi Zou[†2], Yulun Zhang[3], and Luc Van Gool[3]
The Shenzhen International Graduate School, Tsinghua University[1]
Huawei Noah's Ark Lab[2], ETH Zürich[3]

# Outline

- Background and Motivation
- The Proposed Unsupervised Flow-Aligned Seq2Seq Mode
  - S2SVR:  Overall framework
  - Unsupervised Optical Flow Method
- Experiment Results

# Video Restoration

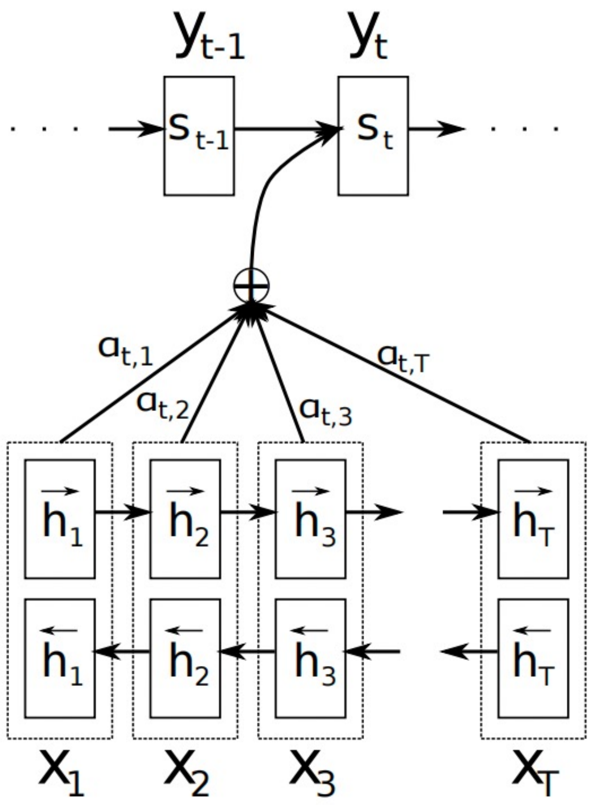



Existing Methods

- CNN-based Methods: utilize information within a short temporal windows, omittance of distant frames
- RNN-based Methods: vanishing gradients problem
- Transformer-based Methods: quadratic complexity to the token number, thus can not capture long-term dependencies

Existing methods show limitations in efficiently modeling the inter-frame relation within the video sequence.

# Seq2Seq Model

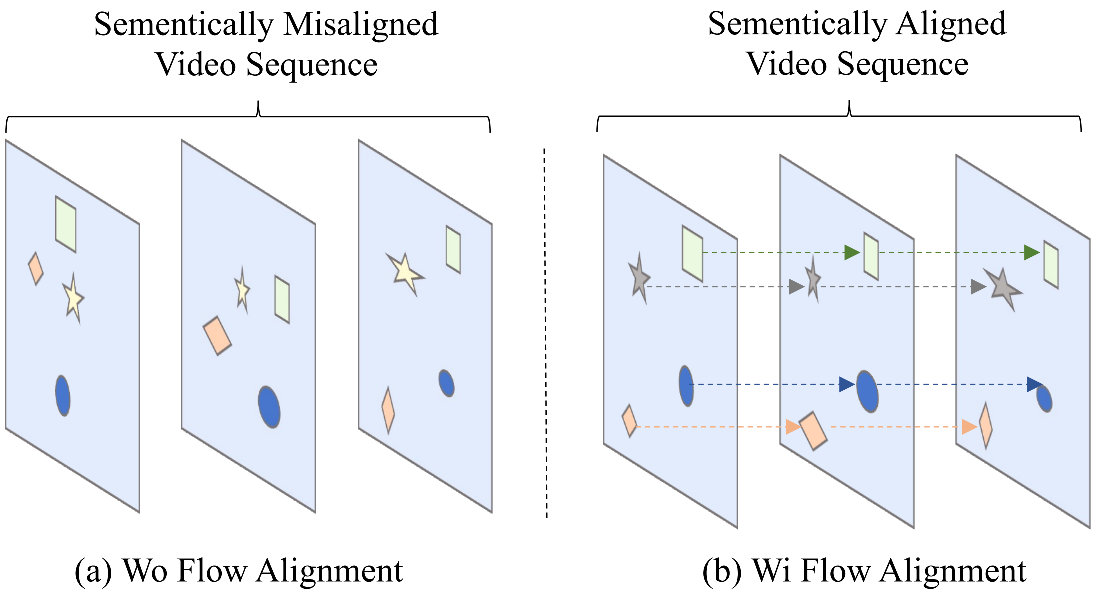



- Seq2Seq model has been proven powerful of sequence modeling in NLP
- An encoder reads and encodes a source sentence into latent representations. A decoder and attention module then outputs a translation from the encoded vector.
- Will Seq2Seq model work in video restoration ?

  Yes: the sequence nature of video signal
  But: the domain difference between NLP and VR:
  1D words & 2D misaligned frames

Flow-Aligned Seq2Seq Model for Video Restoration

# Optical Flow

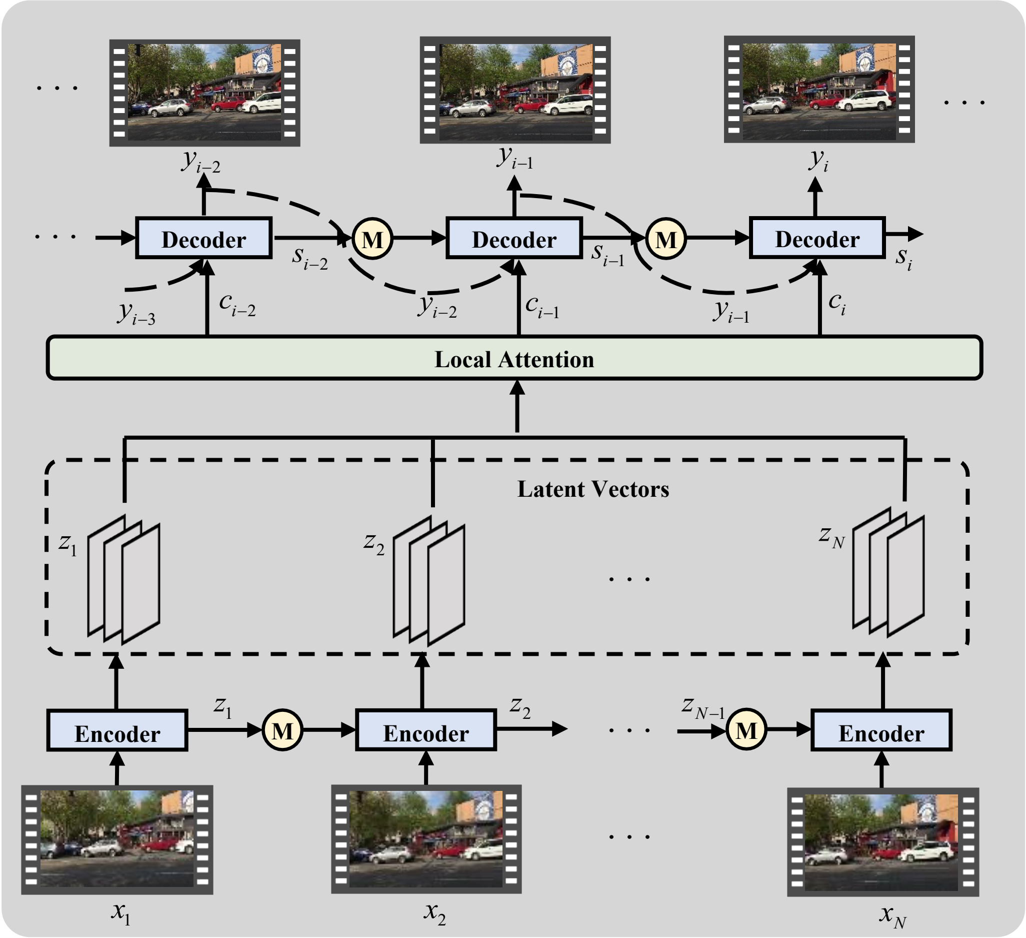



Use optical flow to align the frames and establish accurate sematic correspondence along the sequence, but:

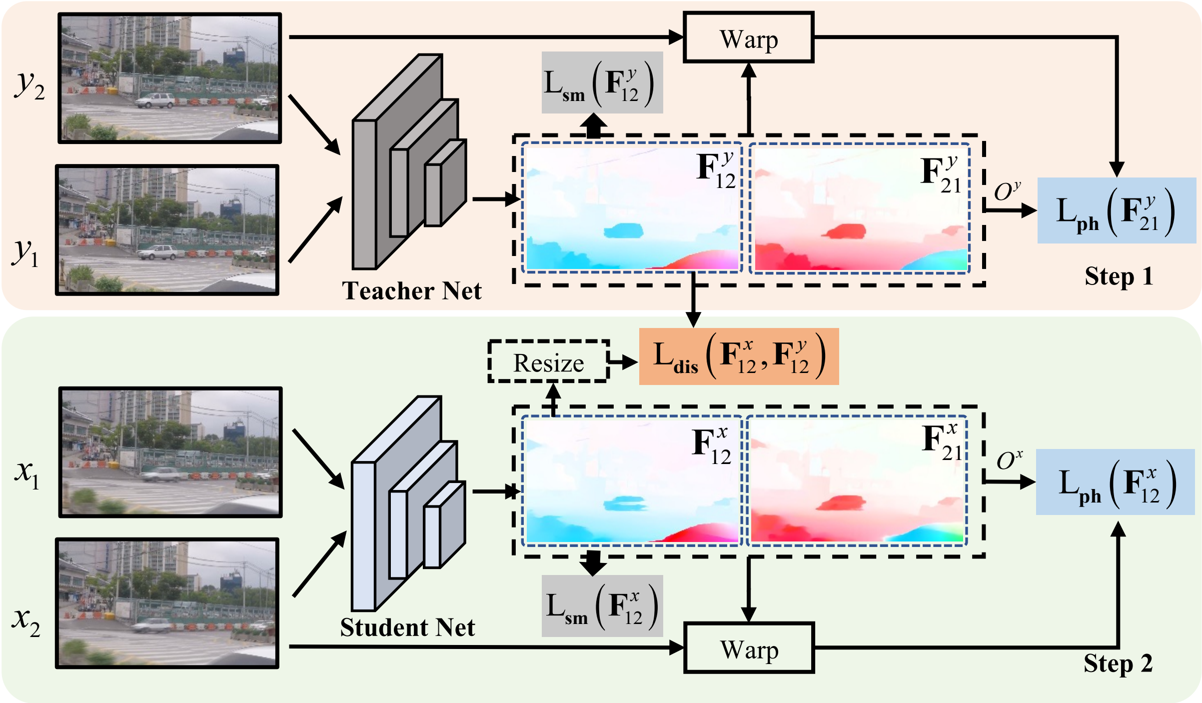
- The flow estimator is pretrained on synthetic flow dataset, will it work in real-world dataset ?
- How to estimate accurate flow from the severely degraded videos ?

➡ Unsupervised Optical Flow Method

# Method




S2SVR




Unsupervised Optical Flow Method

- The first sequence-to-sequence model for video restoration, composed of an encoder, a decoder, and local attention.
- Train with the photometric loss, smooth loss and distillation loss. The data distillation loss is based on the LQ-HQ characteristic of video restoration task.

# Experiments

| QP | Approach | | AR-CNN (Dong et al., 2015) | DnCNN (Zhang et al., 2017) | DS-CNN (Yang et al., 2018a) | MFQE 1.0 (Yang et al., 2018b) | MFQE 2.0 (Guan et al., 2019) | STDF-R3L (Deng et al., 2020) | **S2SVR (Ours)** |
|---|---|---|---|---|---|---|---|---|---|
| | Metrics | | ΔPSNR / ΔSSIM | | | | | | |
| 37 | A | *Traffic* | 0.239 / 47 | 0.238 / 57 | 0.286 / 60 | 0.497 / 90 | 0.585 / 102 | 0.730 / 115 | **0.851 / 138** |
| | | *PeopleOnStreet* | 0.346 / 75 | 0.414 / 82 | 0.416 / 85 | 0.802 / 137 | 0.920 / 157 | 1.250 / 196 | **1.385 / 216** |
| | B | *Kimono* | 0.219 / 65 | 0.244 / 75 | 0.249 / 75 | 0.495 / 113 | 0.550 / 118 | 0.850 / 161 | **1.055 / 195** |
| | | *ParkScene* | 0.136 / 38 | 0.141 / 50 | 0.153 / 50 | 0.391 / 103 | 0.457 / 123 | 0.590 / 147 | **0.649 / 165** |
| | | *Cactus* | 0.190 / 38 | 0.195 / 48 | 0.239 / 58 | 0.439 / 88 | 0.501 / 100 | 0.770 / 138 | **0.828 / 152** |
| | | *BQTerrace* | 0.195 / 28 | 0.201 / 38 | 0.257 / 48 | 0.270 / 48 | 0.403 / 67 | 0.630 / 106 | **0.654 / 115** |
| | | *BasketballDrive* | 0.229 / 55 | 0.251 / 58 | 0.282 / 65 | 0.406 / 80 | 0.465 / 83 | 0.750 / 123 | **0.972 / 157** |
| | C | *RaceHorses* | 0.219 / 43 | 0.253 / 65 | 0.267 / 63 | 0.340 / 55 | 0.394 / 80 | 0.550 / 135 | **0.854 / 203** |
| | | *BQMall* | 0.275 / 68 | 0.281 / 68 | 0.330 / 80 | 0.507 / 103 | 0.618 / 120 | 0.990 / 180 | **1.080 / 205** |
| | | *PartyScene* | 0.107 / 38 | 0.131 / 48 | 0.174 / 58 | 0.217 / 73 | 0.363 / 118 | **0.680** / 194 | 0.628 / **236** |
| | | *BasketballDrill* | 0.247 / 58 | 0.331 / 68 | 0.352 / 68 | 0.477 / 90 | 0.579 / 120 | 0.790 / 149 | **0.949 / 179** |
| | D | *RaceHorses* | 0.268 / 55 | 0.311 / 73 | 0.318 / 75 | 0.507 / 113 | 0.594 / 143 | 0.830 / 208 | **1.010 / 237** |
| | | *BQSquare* | 0.080 / 8 | 0.129 / 18 | 0.201 / 38 | -0.010 / 15 | 0.337 / 65 | 0.640 / 125 | **0.886 / 141** |
| | | *BlowingBubbles* | 0.164 / 35 | 0.184 / 58 | 0.228 / 68 | 0.386 / 120 | 0.533 / 170 | **0.740** / 226 | 0.710 / **230** |
| | | *BasketballPass* | 0.259 / 58 | 0.307 / 75 | 0.335 / 78 | 0.628 / 138 | 0.728 / 155 | 1.080 / 212 | **1.110 / 222** |
| | E | *FourPeople* | 0.373 / 50 | 0.388 / 60 | 0.459 / 70 | 0.664 / 85 | 0.734 / 95 | 0.940 / 117 | **1.021 / 136** |
| | | *Johnny* | 0.247 / 10 | 0.315 / 40 | 0.378 / 40 | 0.548 / 55 | 0.604 / 68 | 0.810 / 88 | **0.976 / 120** |
| | | *KristenAndSara* | 0.409 / 50 | 0.421 / 60 | 0.481 / 60 | 0.655 / 75 | 0.754 / 85 | 0.970 / 96 | **1.035 / 113** |
| | Average | | 0.233 / 45 | 0.263 / 58 | 0.300 / 63 | 0.455 / 88 | 0.562 / 109 | 0.830 / 151 | **0.925 / 176** |

Tab. 1 Comparison with SOTA methods on video quality enhancement dataset.

Our S2SVR outperforms SOTA methods on three video restoration tasks.

| Methods | Params | REDS4 | Vimeo-90K-T |
|---|---|---|---|
| Bicubic | - | 26.14 / 0.7292 | 31.32 / 0.8684 |
| TOFlow | - | 27.98 / 0.7990 | 33.08 / 0.9054 |
| DUF | 5.8 M | 28.63 / 0.8251 | - |
| RBPN* | 12.2 M | 30.09 / 0.8590 | 37.07 / 0.9435 |
| EDVR-M | 3.3 M | 30.53 / 0.8699 | 37.09 / 0.9446 |
| EDVR | 20.6 M | 31.09 / 0.8800 | 37.61 / 0.9489 |
| PFNL | 3.0 M | 29.63 / 0.8502 | 36.14 / 0.9363 |
| MuCAN | - | 30.88 / 0.8750 | 37.32 / 0.9465 |
| BasicVSR | 6.3 M | 31.42 / 0.8909 | 37.18 / 0.9450 |
| IconVSR | 8.7 M | 31.67 / 0.8948 | 37.47 / 0.9476 |
| VSR-Transformer | 32.6 M | 31.19 / 0.8815 | **37.71 / 0.9494** |
| **S2SVR (Ours)** | 13.4 M | **31.96 / 0.8988** | 37.63 / 0.9490 |

Tab. 2 Video super-resolution results.

| Methods | Params | PSNR (dB) | SSIM |
|---|---|---|---|
| Tao *et al.* | - | 30.29 | 0.9014 |
| Su *et al.* | 15.30 M | 27.31 | 0.8255 |
| Kim *et al.* | - | 26.82 | 0.8245 |
| Nah *et al.* | - | 29.97 | 0.8947 |
| EDVR | 23.6 M | 26.83 | 0.8426 |
| STFAN | 5.37 M | 28.59 | 0.8608 |
| TSP | 16.19 M | 31.67 | **0.9279** |
| UHDVD | - | 31.33 | 0.9210 |
| **S2SVR (Ours)** | 8.44 M | **31.81** | 0.9231 |

Tab. 3 Video deblurring results.

# Experiments


Fig. 1 Quality comparison with SOTA methods on REDS4 dataset.

# Thanks

VR-Baseline

Code & Paper