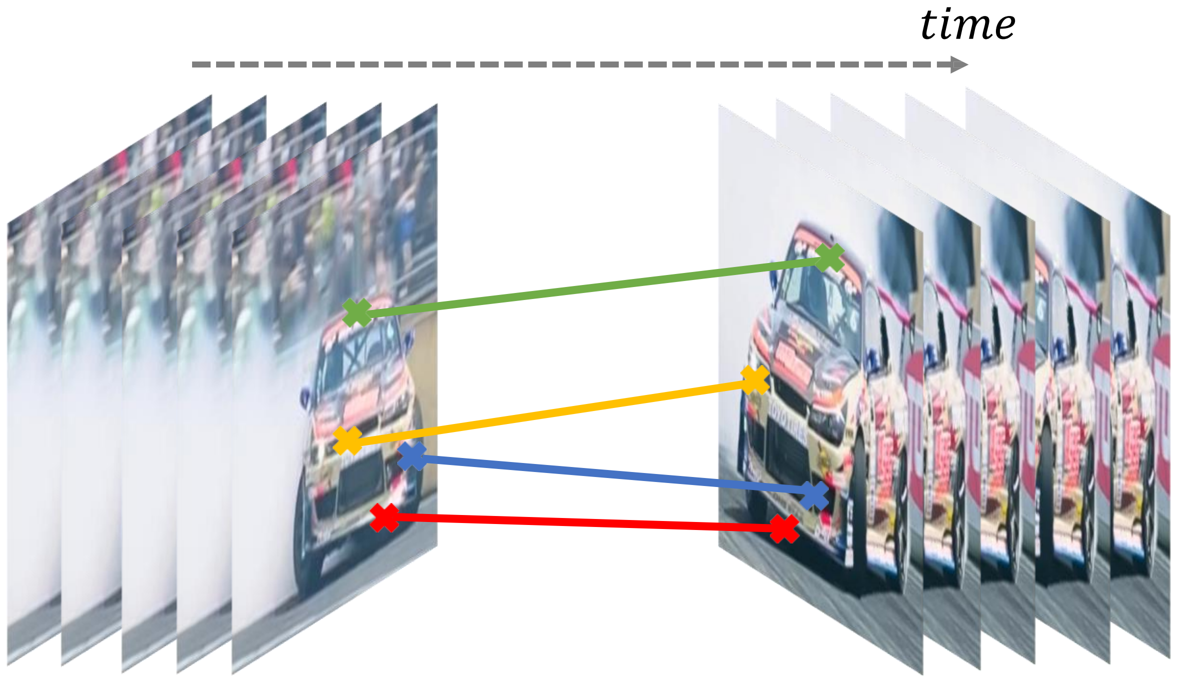
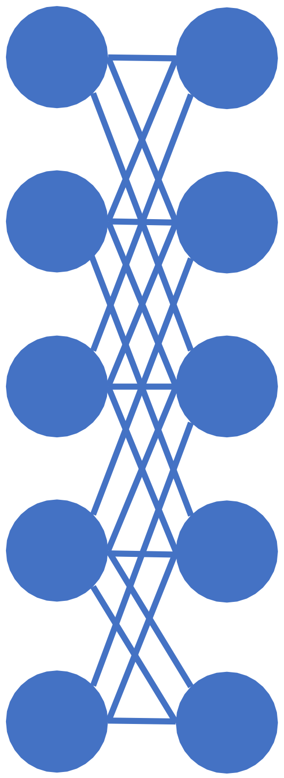
Self-supervised Learning for Temporal Correspondence

CVPR 2020 Tutorial

Learning Inter-Frame Relations



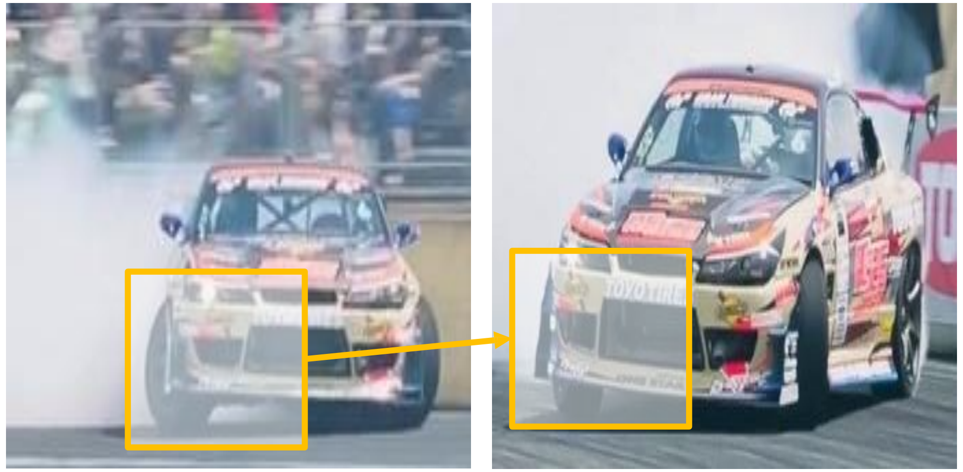


[1] Xueting Li, Sifei Liu, et al. Joint-task Self-supervised Learning for Temporal Correspondence. In *NIPS*, 2019

Motivation

Determine a bbox in each frame:
1.Tracking-by-detection frames independently
2.Tracking-by-matching framework (this work)

- Region-level matching: tracking large image regions between consecutive video frames.

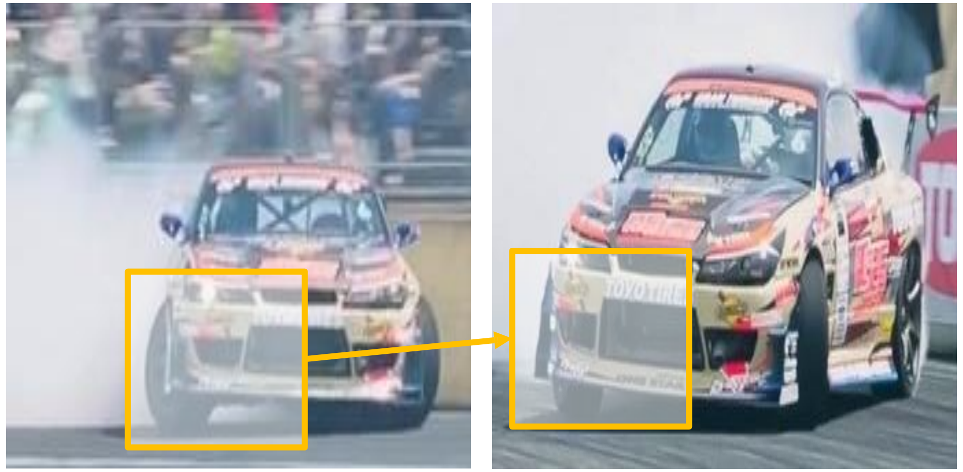


(a) Region-level matching

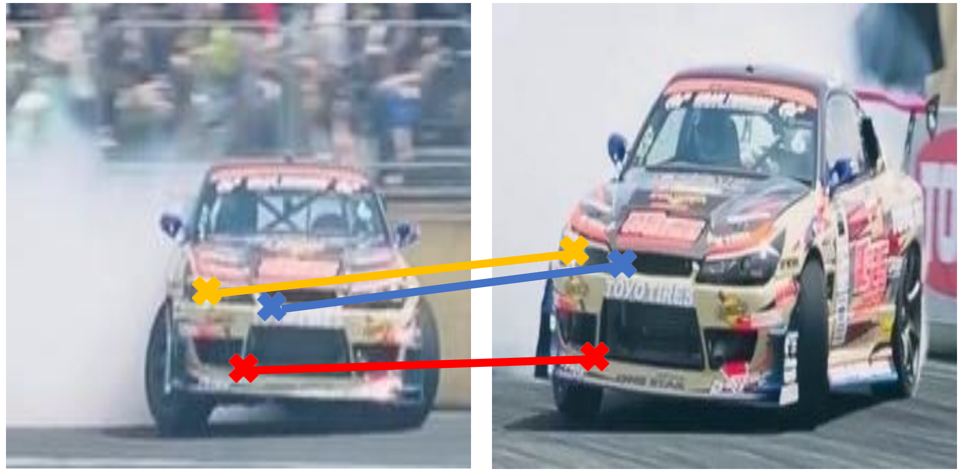
Motivation

Track individual pixels:
Direct regression of pixel offsets

- Region-level matching: tracking large image regions between consecutive video frames.
- Fine-grained matching: establishing fine-grained pixel-level associations between consecutive video frames.



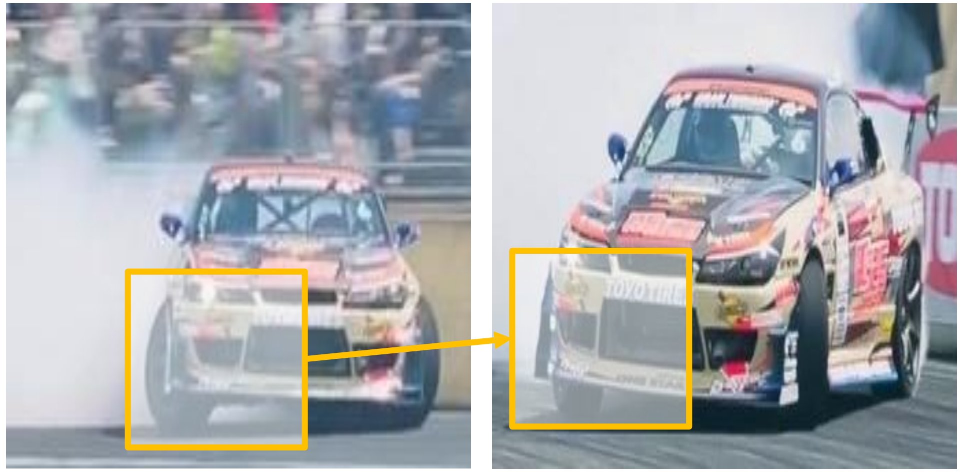
(a) Region-level matching



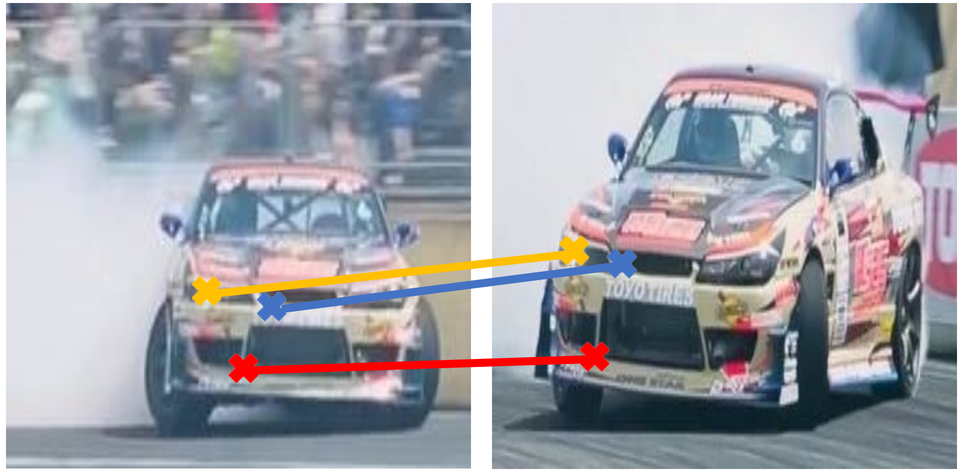
(b) Fine-grained matching

Motivation

- Region-level matching: tracking large image regions between consecutive video frames.
- Fine-grained matching: establishing fine-grained pixel-level associations between consecutive video frames.
- Datasets with annotations for both tasks are scarcely available.



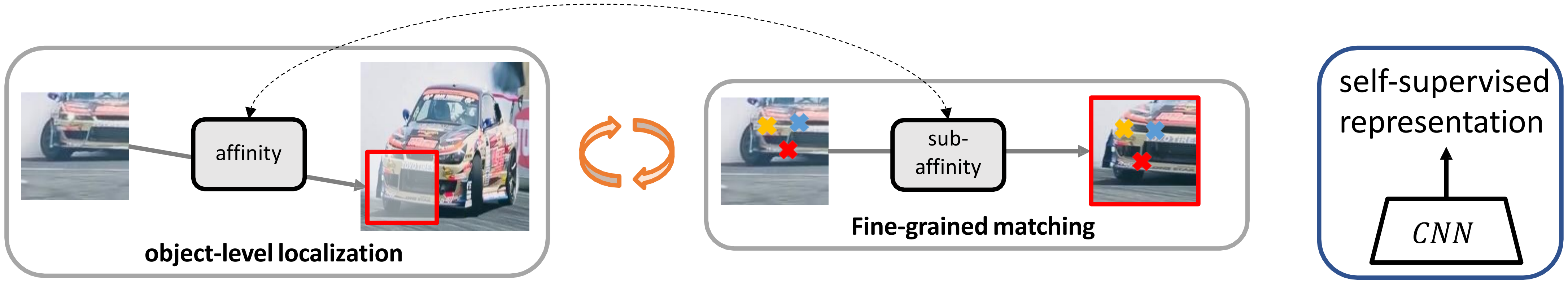
(a) Region-level matching



(b) Fine-grained matching

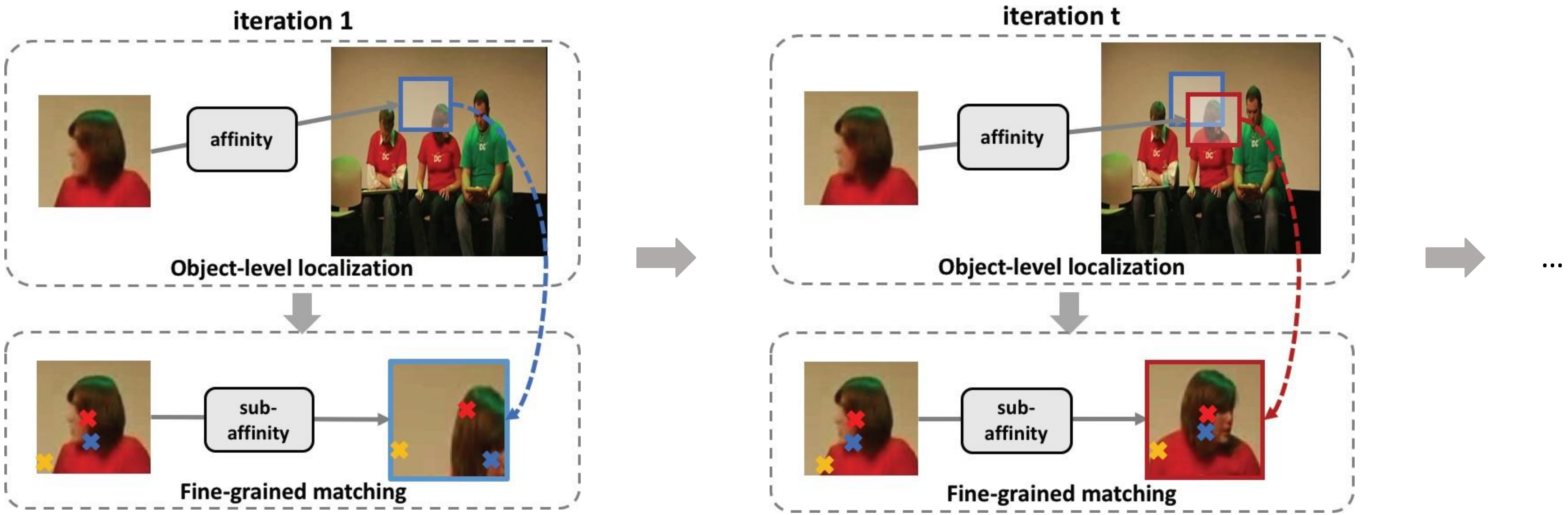
Motivation

- We exploit the synergy between both tasks through a shared inter-frame affinity matrix, which simultaneously models transitions between video frames at both the region- and pixel-levels.
- Region-level module: finds a pair of patches with matching parts in the two frames.
- Fine-grained module: reconstructs the color feature by transforming it between the patches.
- Self-supervised: using the ground-truth color as the self-supervisory signal. (datasets problem)



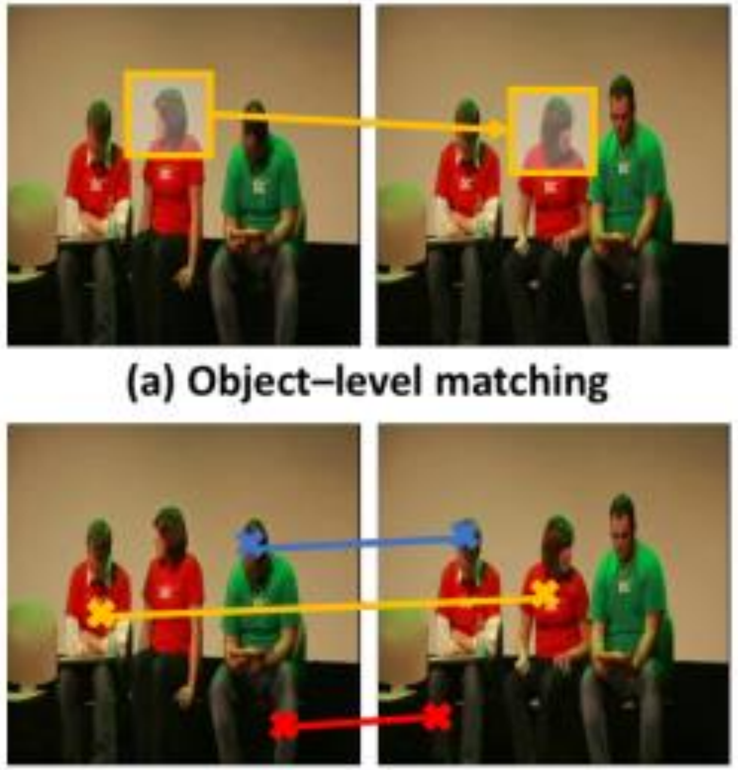
Motivation

- Region-level localization helps reduce ambiguities in fine-grained matching by narrowing down search regions.
- Fine-grained matching provides bottom-up features to facilitate region-level localization.



Motivation

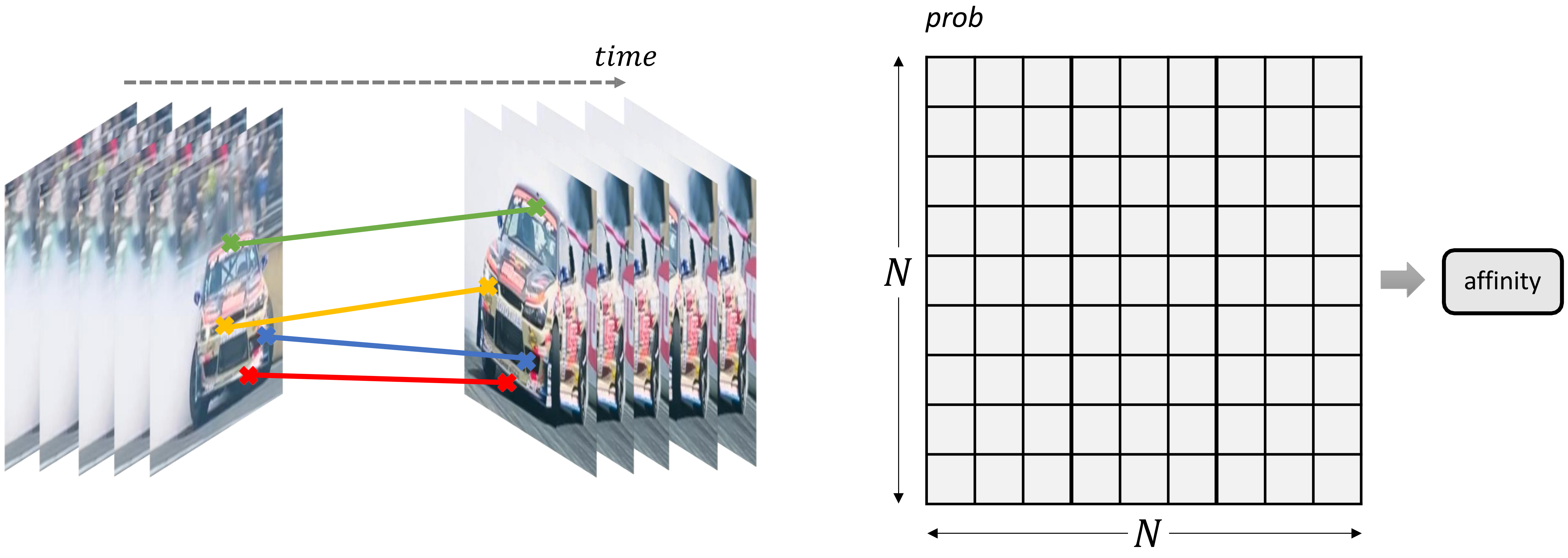
- Region-level localization helps reduce ambiguities in fine-grained matching by narrowing down search regions.



(a) Object–level matching

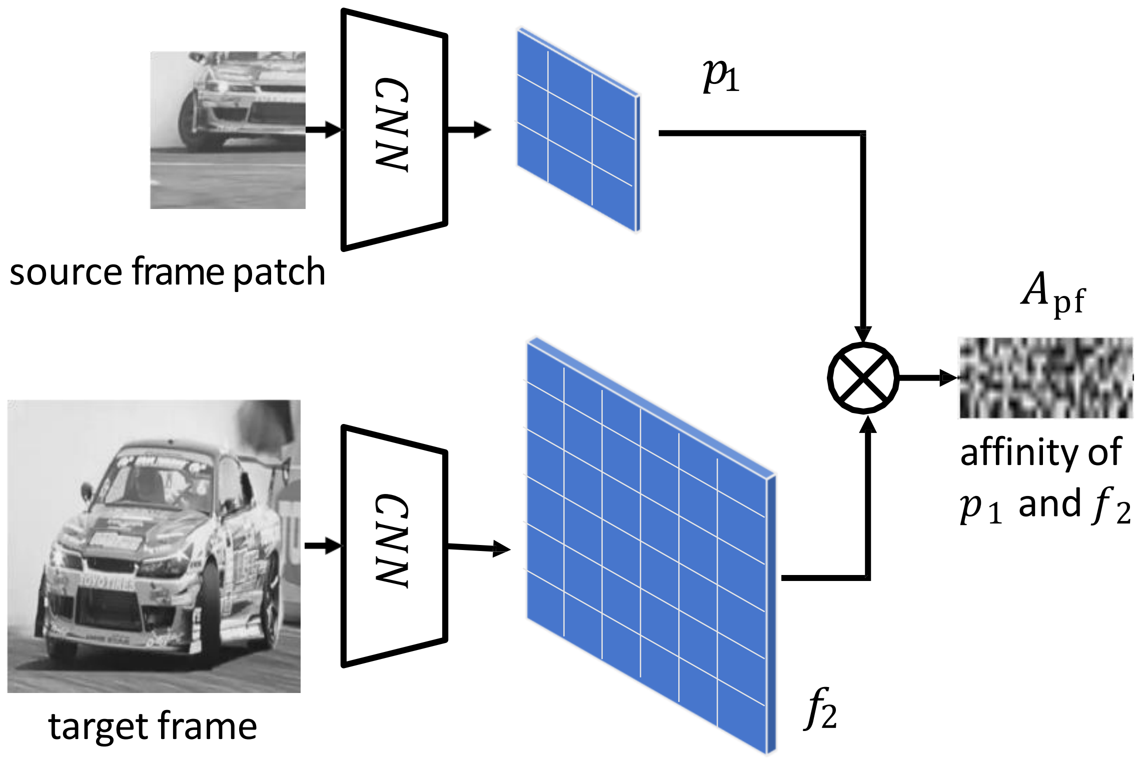
(b) Fine-grained matching

Transforming Feature and Location via Affinity



Framework

Gray-scale image as input



Affinity matrix: $A_{ij} = 1$ Directly copy from 1st frame ith pixel to 2nd frame jth pixel

$$A_{ij} = \kappa(f_{1i}, f_{2j}) \qquad f_1 \in \mathcal{R}^{C \times N_1} \quad f_2 \in \mathcal{R}^{C \times N_2}$$

$$A_{ij} = \frac{\exp(f_{1i}^\top f_{2j})}{\sum_k \exp(f_{1k}^\top f_{2j})}, \quad \forall i \in [1, N_1], j \in [1, N_2]$$

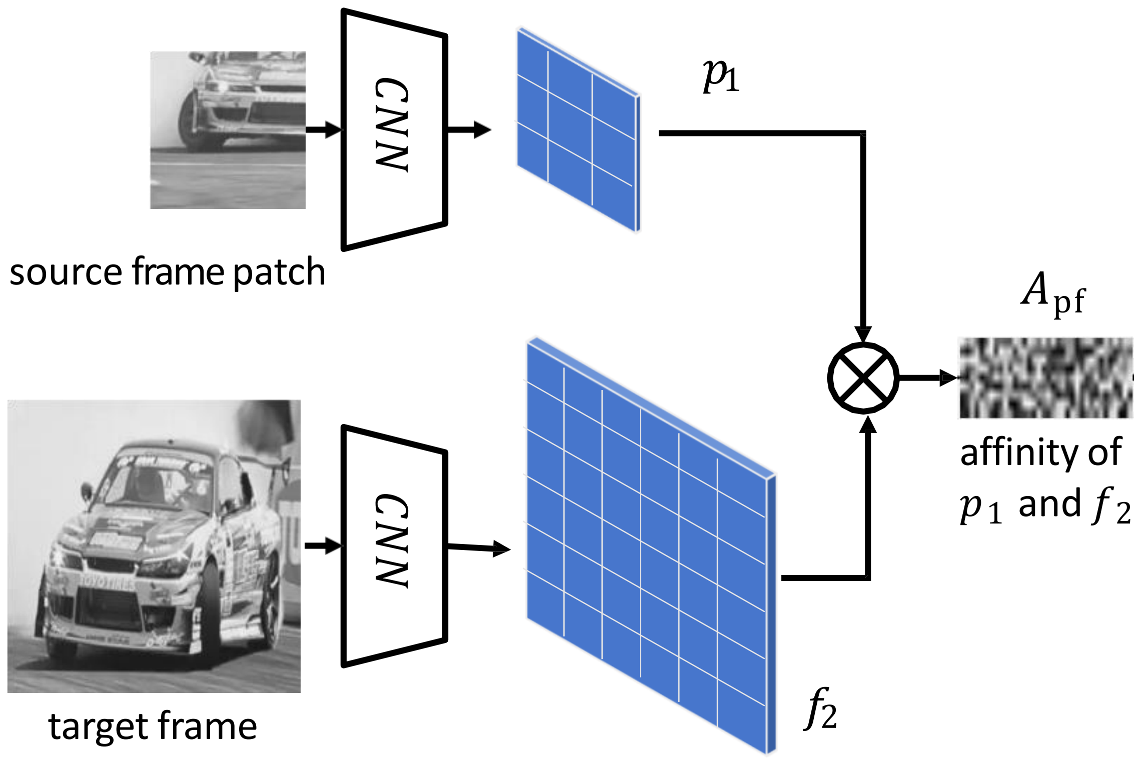
1. Affinity matrix between patch p_1 and frame f_2 is computed as a dot product.

2. Each column is the similarity score between a point in the target frame to all points in the reference frame.

$\otimes$ matrix multiplication

Framework

Gray-scale image as input



Affinity matrix: $A_{ij} = 1$ Directly copy from 1st frame ith pixel to 2nd frame jth pixel

$$A_{ij} = \kappa(f_{1i}, f_{2j}) \qquad f_1 \in \mathcal{R}^{C \times N_1} \quad f_2 \in \mathcal{R}^{C \times N_2}$$

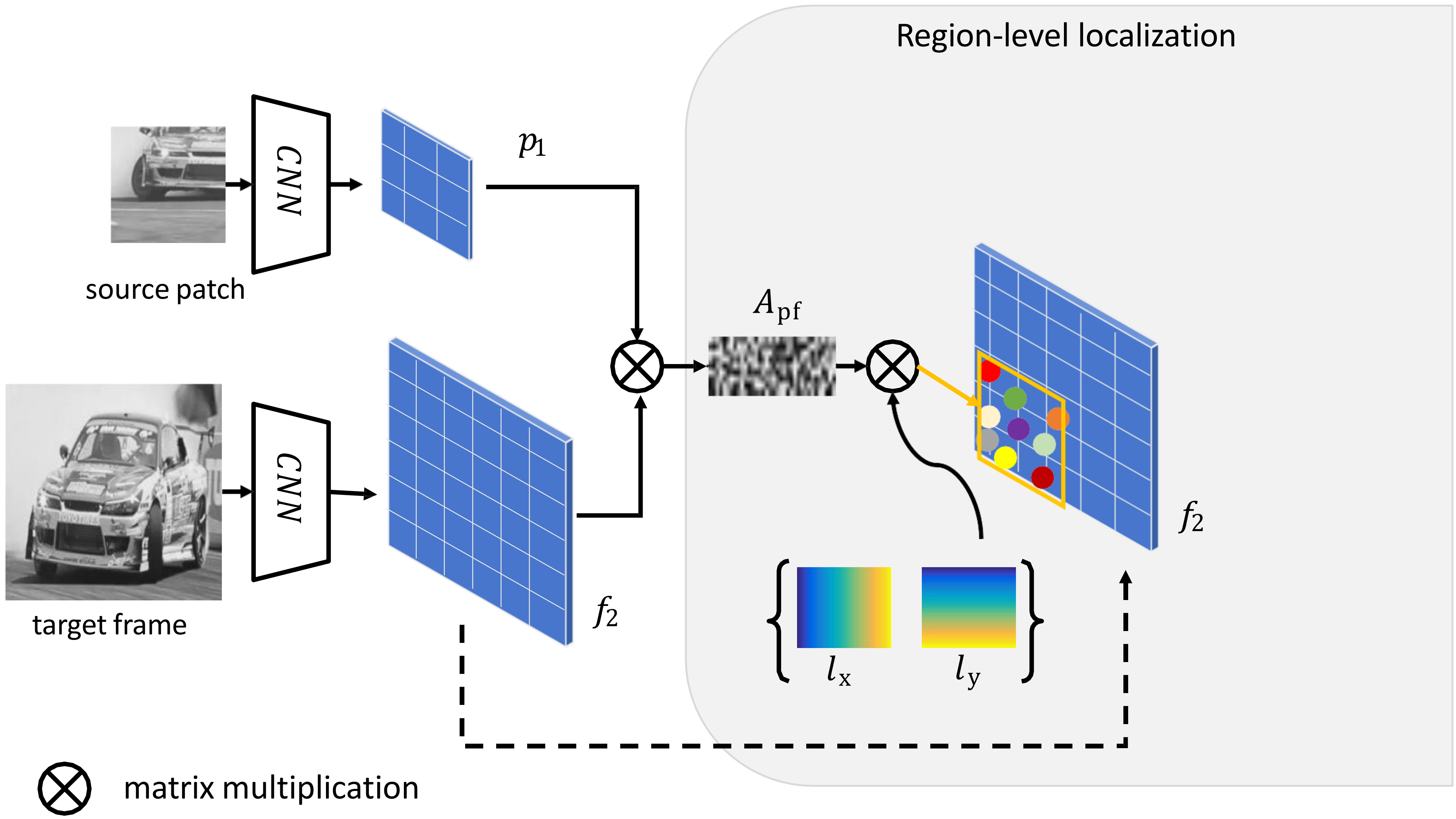
$$A_{ij} = \frac{\exp(\boxed{f_{1i}^\top f_{2j}})}{\sum_k \exp(f_{1k}^\top f_{2j})}, \quad \forall i \in [1, N_1], j \in [1, N_2]$$

1. Affinity matrix between patch p_1 and frame f_2 is computed as a dot product.

2. Each column is the similarity score between a point in the target frame to all points in the reference frame.

$\otimes$ matrix multiplication

Framework



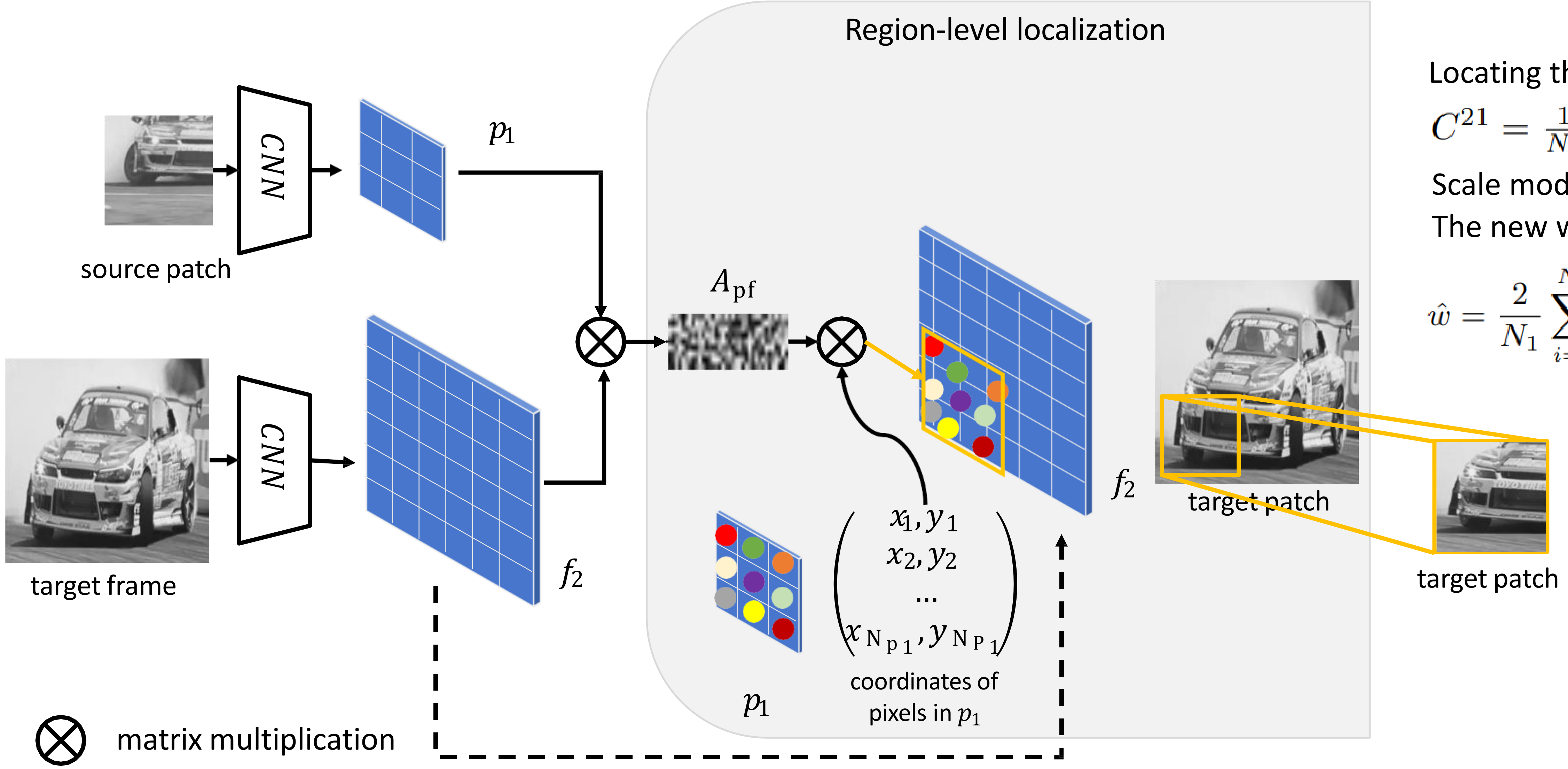
Vectorized location map for N pixels:

$$l_j = (x_j, y_j), l \in \mathcal{R}^{2\times N}$$

The location of pixel traced from reference patch to target frame:

$$l_j^{12} = \sum_{k=1}^{N_1} l_k^{11} A_{kj}, \quad \forall j \in [1, N_2]$$

Framework



Locating the center of the target patch:

$$C^{21} = \frac{1}{N_1}\sum_{i=1}^{N_1} l_i^{21}$$

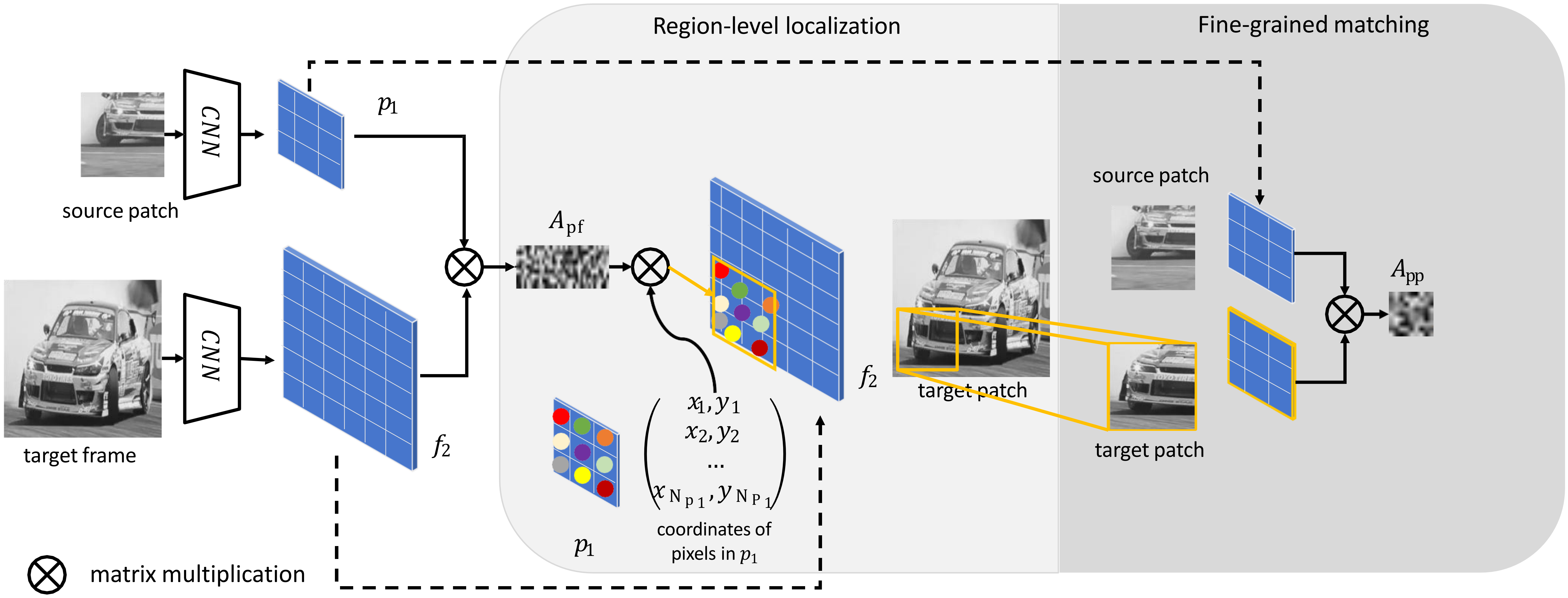
Scale modeling:

The new width w of the new bbox:

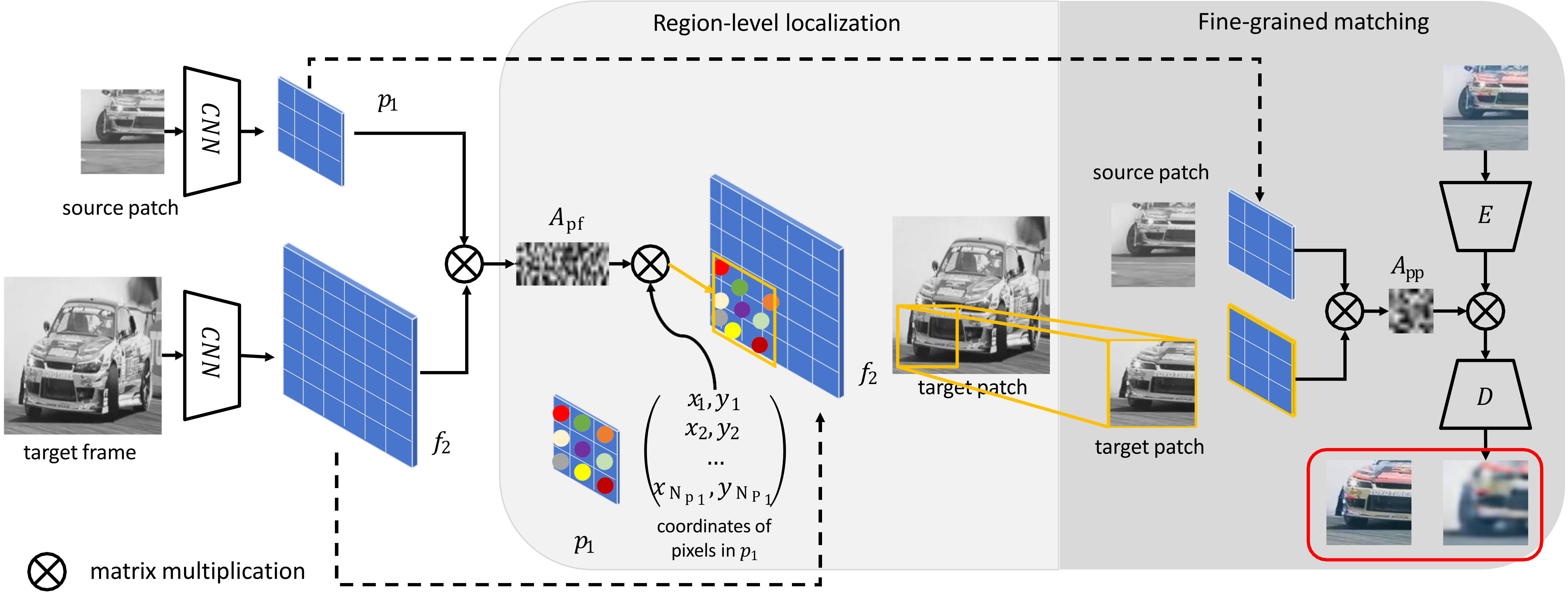
$$\hat{w} = \frac{2}{N_1}\sum_{i=1}^{N_1}\left\|x_i - C^{21}(x)\right\|_1$$

Framework

Sub-affinity matrix A_{pp}
containing the columns corresponding to the located pixels in the target patch

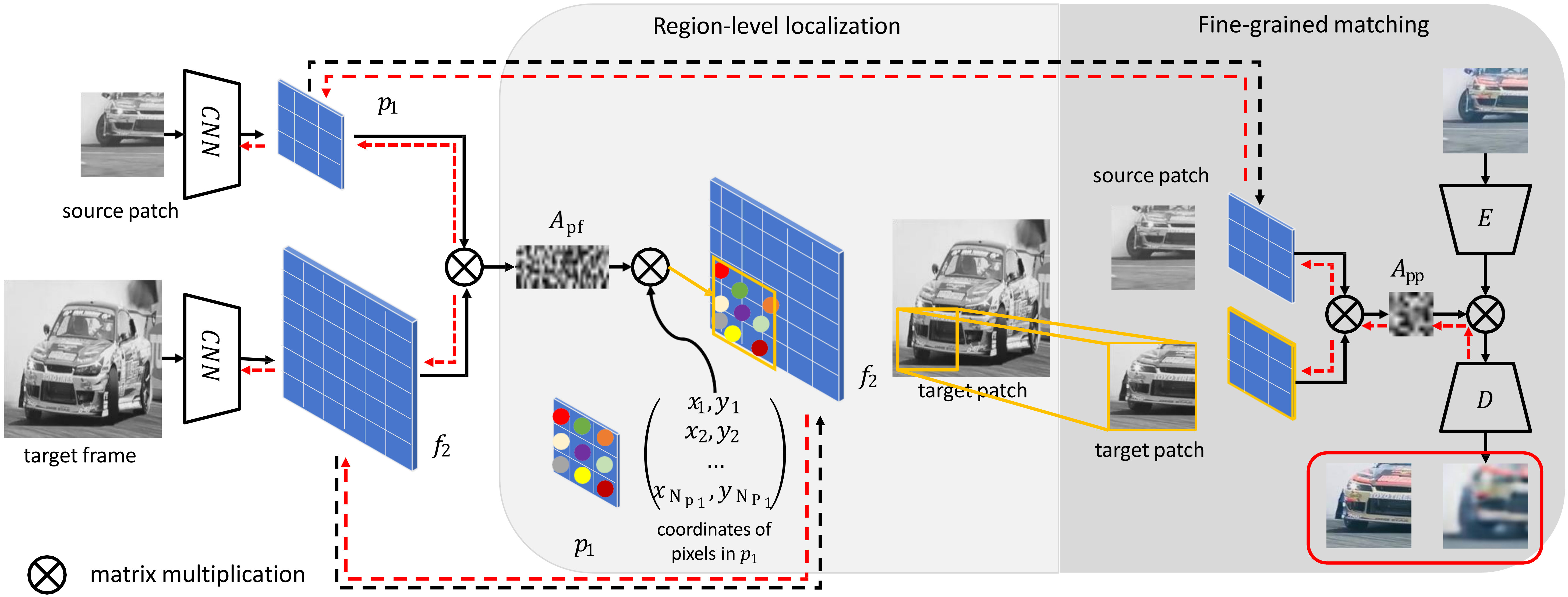


Framework

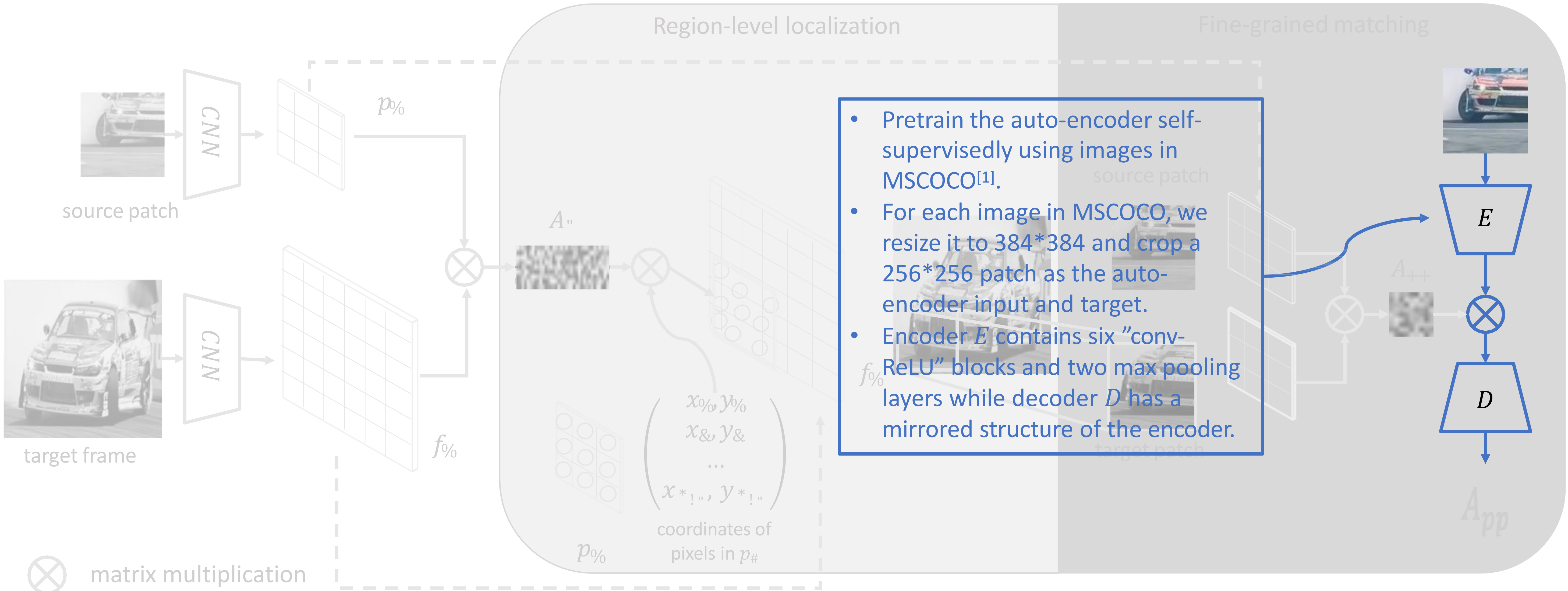


Region-level localization
Fine-grained matching
source patch
CNN
p_1
target frame
CNN
f_2
A_{pf}
f_2
p_1
x_1, y_1
x_2, y_2
...
$x_{N_{p1}}, y_{N_{p1}}$
coordinates of pixels in p_1
target patch
source patch
target patch
E
A_{pp}
D
matrix multiplication

Framework


Region-level localization
Fine-grained matching
CNN
CNN
p_1
source patch
target frame
f_2
A_{pf}
f_2
p_1
x_1, y_1
x_2, y_2
...
$x_{N_{p1}}, y_{N_{p1}}$
coordinates of pixels in p_1
target patch
source patch
target patch
A_{pp}
E
D
matrix multiplication

Network Training



- Pretrain the auto-encoder self-supervisedly using images in MSCOCO[1].
- For each image in MSCOCO, we resize it to 384*384 and crop a 256*256 patch as the auto-encoder input and target.
- Encoder E contains six "conv-ReLU" blocks and two max pooling layers while decoder D has a mirrored structure of the encoder.

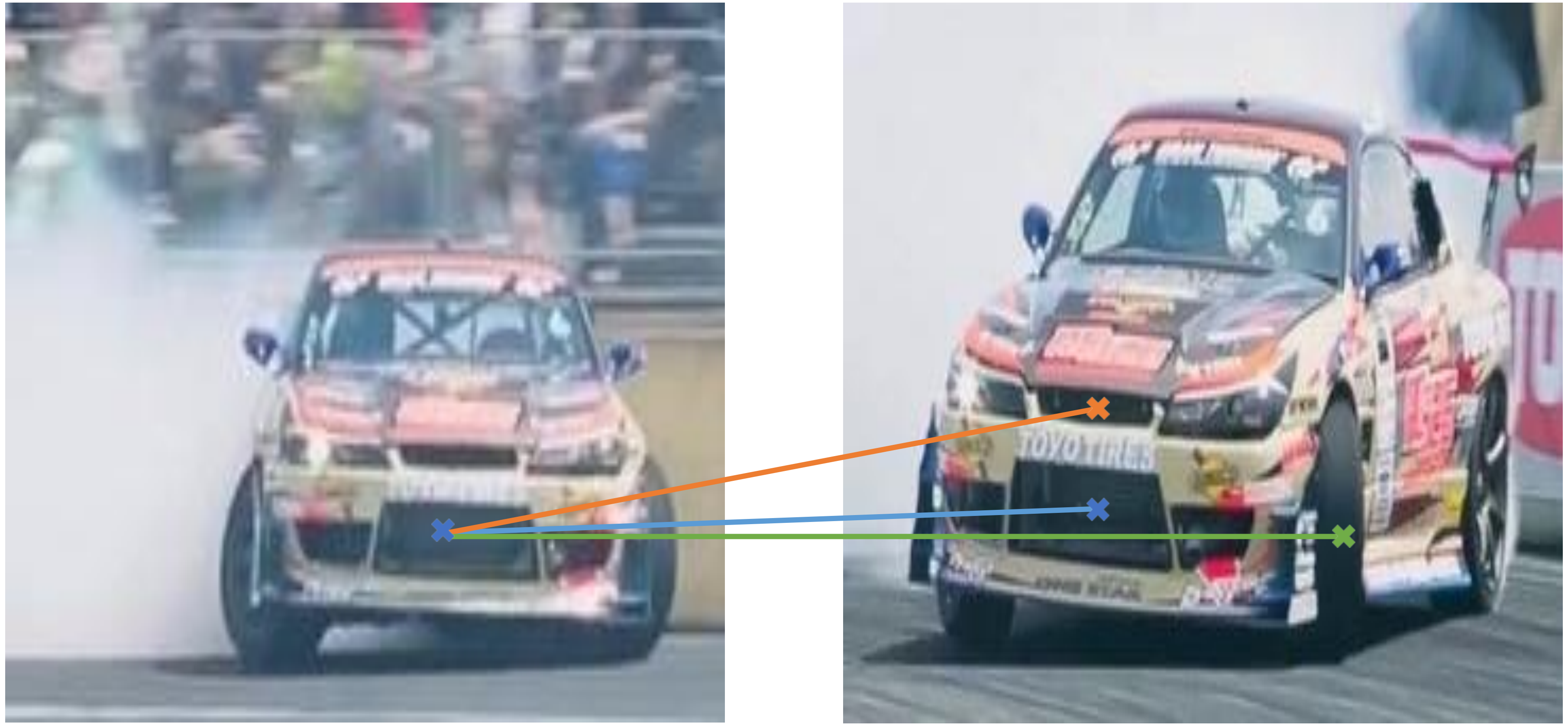
[1] Lin, Tsung-Yi, et al. Microsoft coco: Common objects in context. ECCV, 2014.

Network Training

Then fix the auto-encoder and self-supervisedly train the feature extractor (a ResNet18 with 4 residual blocks) using videos in the Kinetics[1] dataset from scratch.

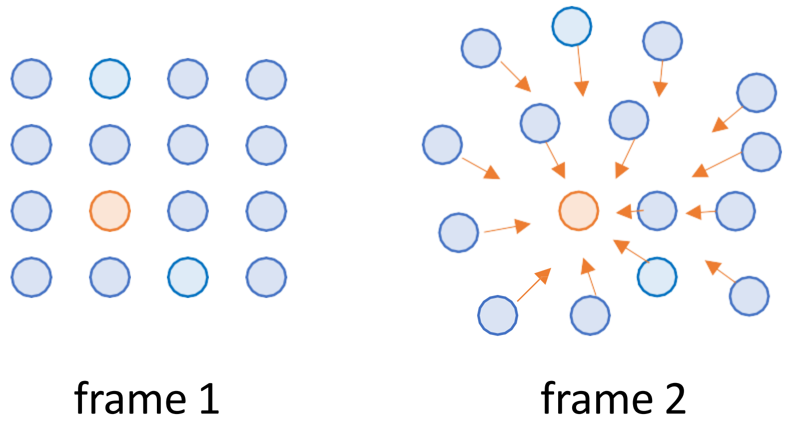
[1] W.Kay, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017

Matching Ambiguities



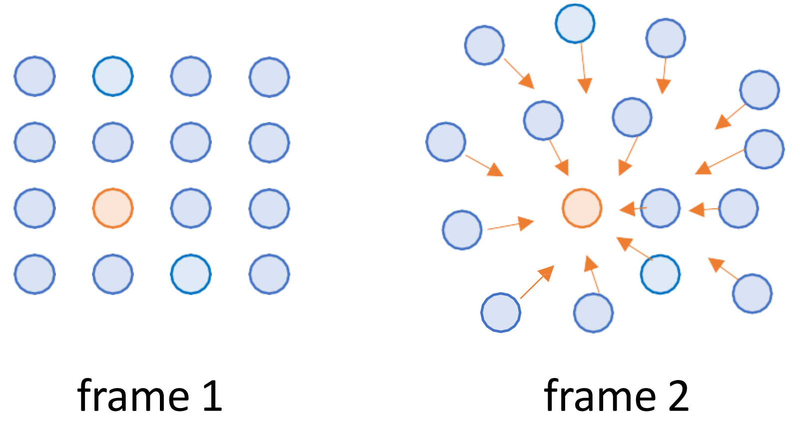
Concentration Regularization

- We constrain that pixels close to each other in the source frame to stay close in the target frame.



Concentration Regularization

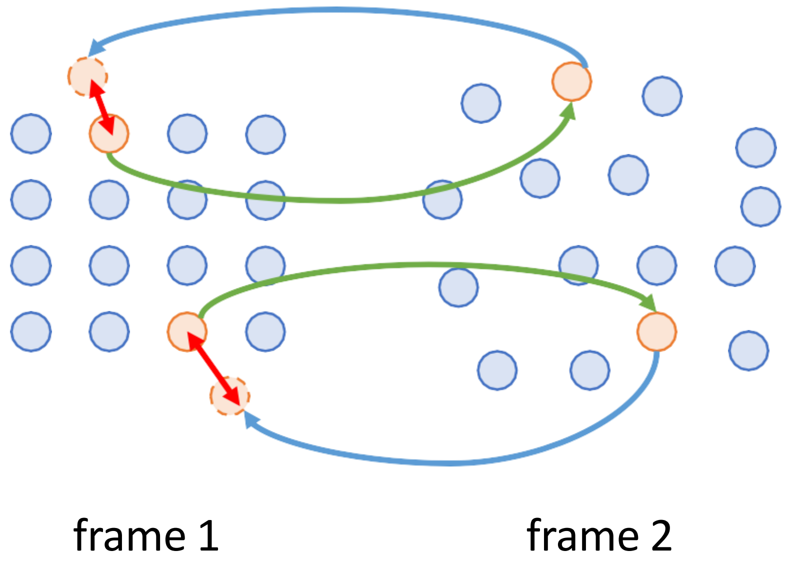
- We constrain that pixels close to each other in the source frame to stay close in the target frame.



$$L_c = \begin{cases} 0, & \left\| l_j^{12}(x) - C^{12}(x) \right\|_1 \leq w \text{ and } \left\| l_j^{12}(y) - C^{12}(y) \right\|_1 \leq h \\ \frac{1}{N_2} \sum_{j=1}^{N_2} \left\| l_j^{12} - C^{12} \right\|_2, & \text{otherwise} \end{cases}$$

Orthogonal Regularization

- For a pair of patches, we encourage every pixel to fall into the same location after one cycle of forward and backward tracking.



By feature matching from frame 1 to frame2:

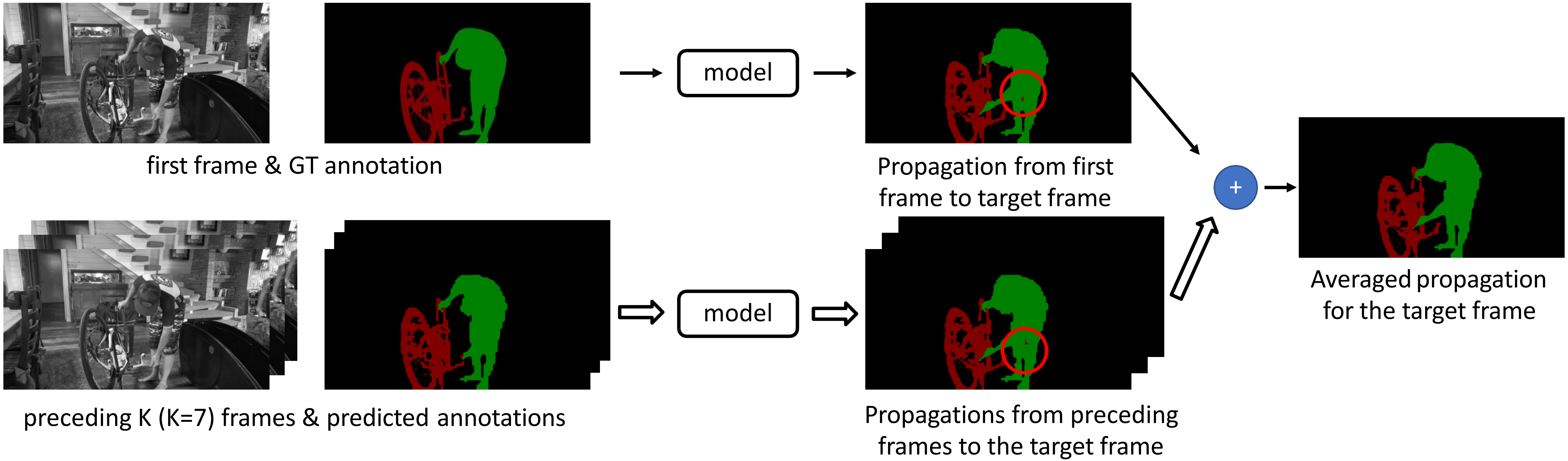
$$\hat{f}_2 = f_1 A_{12}$$
$$\hat{f}_1 = \hat{f}_2 A_{21} = f_1 A_{12} A_{21}$$
$$\boxed{A_{12}^{-1} = A_{21}}$$

By energy preservation between two frames:

$$f_1 f_1^T = f_2 f_2^T = f_1 A_{12} A_{12}^T f_1^T$$
$$A_{12}^{-1} = A_{21} = A_{12}^T$$
$$\boxed{A_{21} = A_{12}^T}$$

Inference

- We use a recursive inference strategy to minimize noise in propagation[1].

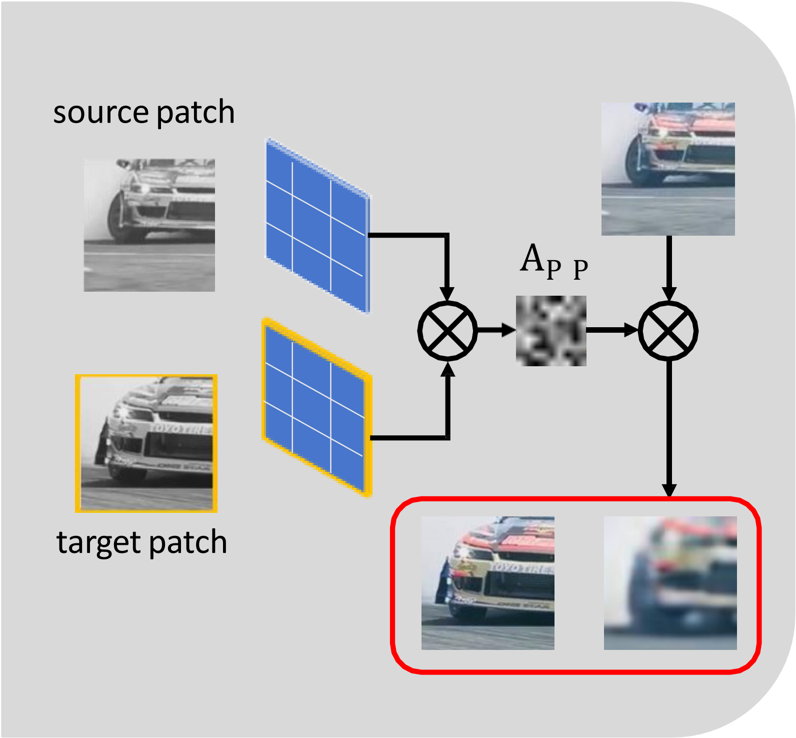


[1] X. Wang, A. Jabri, and A. A. Efros. Learning correspondence from the cycle-consistency of time. In CVPR, 2019

Instance mask propagation on DAVIS-2017

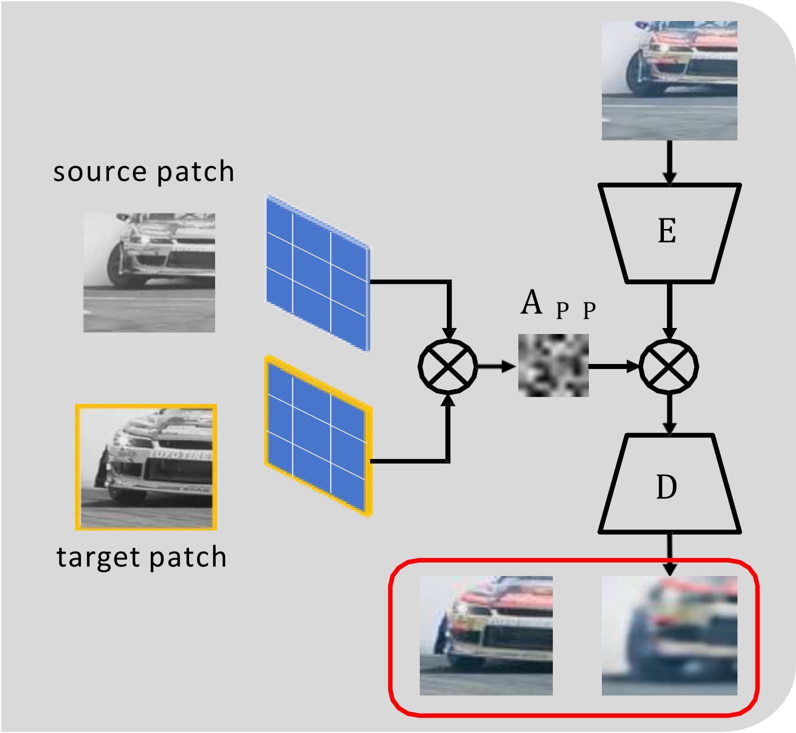
Adding Autoencoder

Vondrick et al. ECCV 2018



J-mean: 34.6

OURS



J-mean: 45.7

C. Vondrick, et al. Tracking emerges by colorizing videos. In *ECCV*, 2018

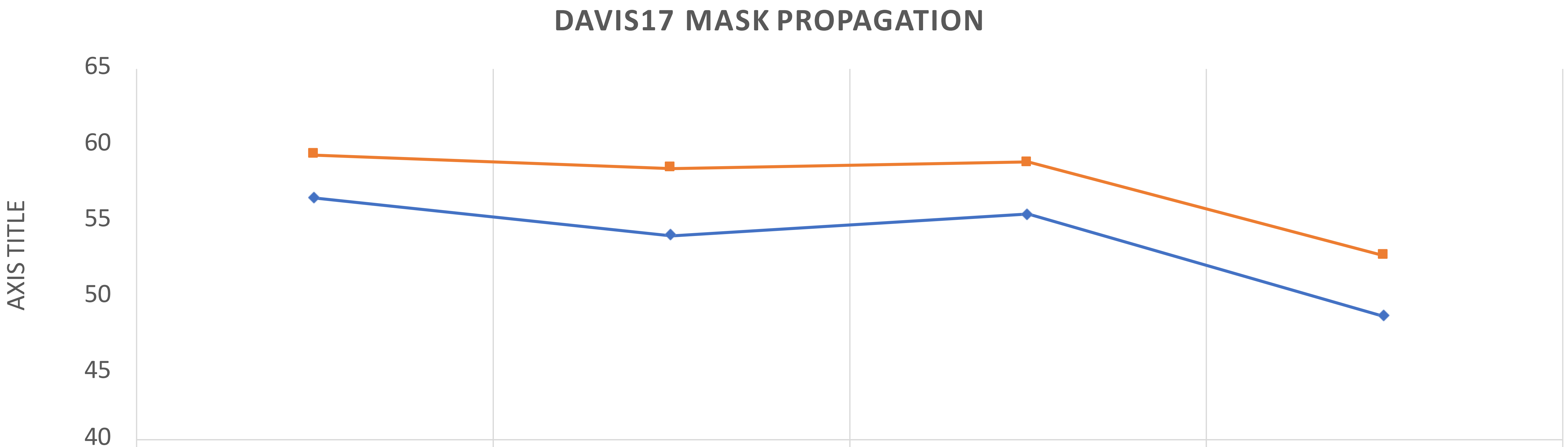
(a) Reference frame (b) ResNet-18 (c) Wang et al. (d) Ours (e) Ours-track (f) Target ground truth

Figure 5: Qualitative comparison with other methods. (a) Reference frame with instance masks. (b) Results by the ResNet-18 trained on ImageNet. (c) Results by Wang et al. [52]. (d) Ours (global matching). (e) Ours with localization during inference. (f) Target frame with ground truth instance masks.

(a) Reference frame (b) Target frame (c) Ours (d) w/o L (e) w/o C (f) w/o O (g) w/o all

Figure 6: Visualization of the ablation studies. Given a set of points in the reference frame (a), we visualize the results of propagating these points on to the target frame (b). “L”, “C”, “O” and “all” correspond to the localization modules, concentration or orthogonal regularization, or all of them (d-g).

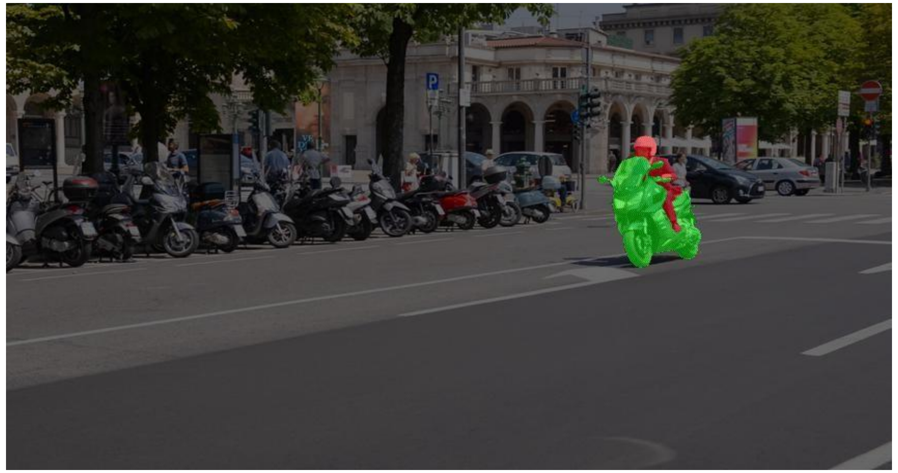
Instance mask propagation on DAVIS-2017



	Ours	wo Localization	wo orthogonal	wo concentration
J-mean	56.3	53.8	55.2	48.3
F-mean	59.2	58.3	58.7	52.4

Results

- Instance mask propagation on DAVIS-2017[1]



Input frame & instance mask

Ours

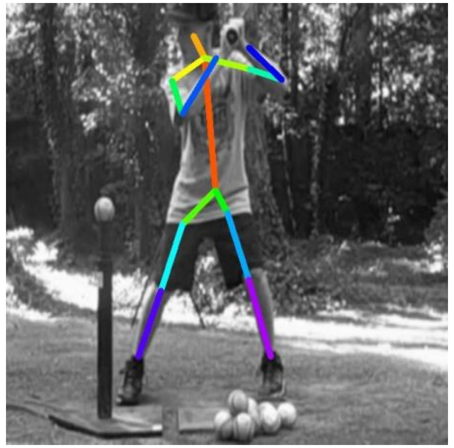
Wang et al.[2]

1 J. Pont-Tuset, et al. The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675, 2017 .

2 X. Wang, A. Jabri, and A. A. Efros. Learning correspondence from the cycle-consistency of time. In CVPR, 2019

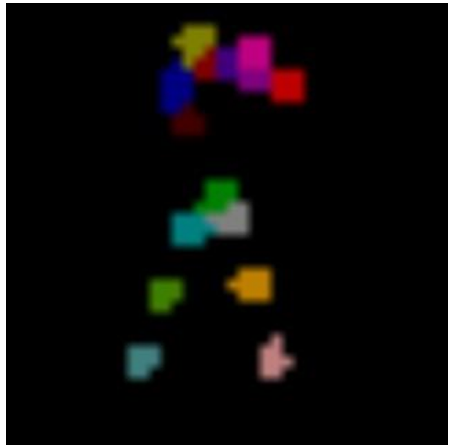
Results

- Pose keypoints propagation on the JHMDB[1] dataset.
- We convert the keypoints of the first frame to a heat map and then propagate the heat map through the rest of video similarly as the segmentation masks.
- We then recover the keypoints from the propagated heat maps by taking the location of maximum response.

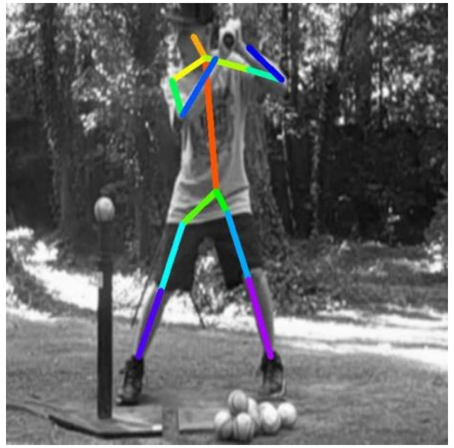


GT pose annotation

heat map



propagated heat map

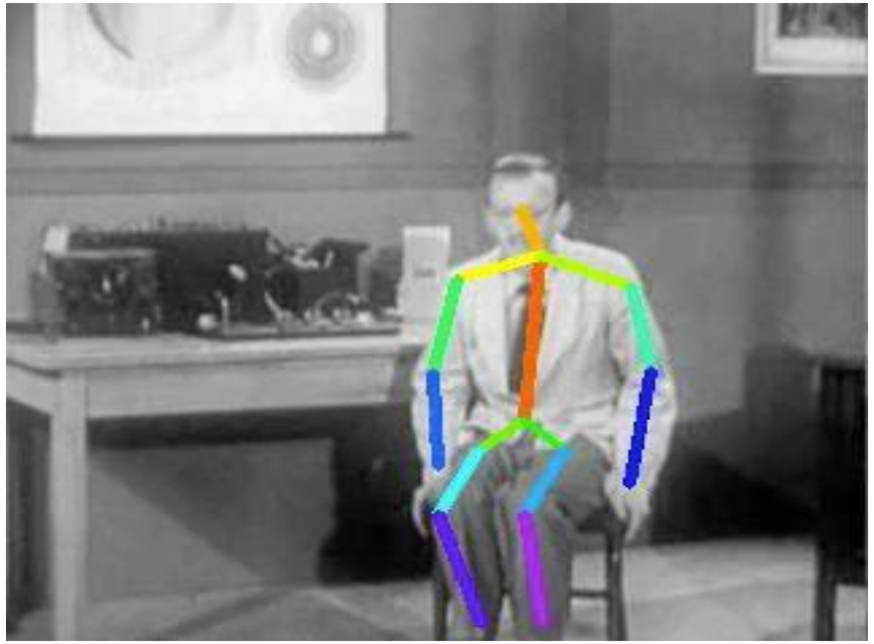


recovered pose

[1] H. Jhuang et al. Towards understanding action recognition. In *ICCV*, 2013

Results

- Pose keypoints propagation on the JHMDB[1] dataset.



[1] H. Jhuang et al. Towards understanding action recognition. In *ICCV*, 2013

Results

- Human parts propagation on the VIP[1] dataset.

Input frame & parts mask

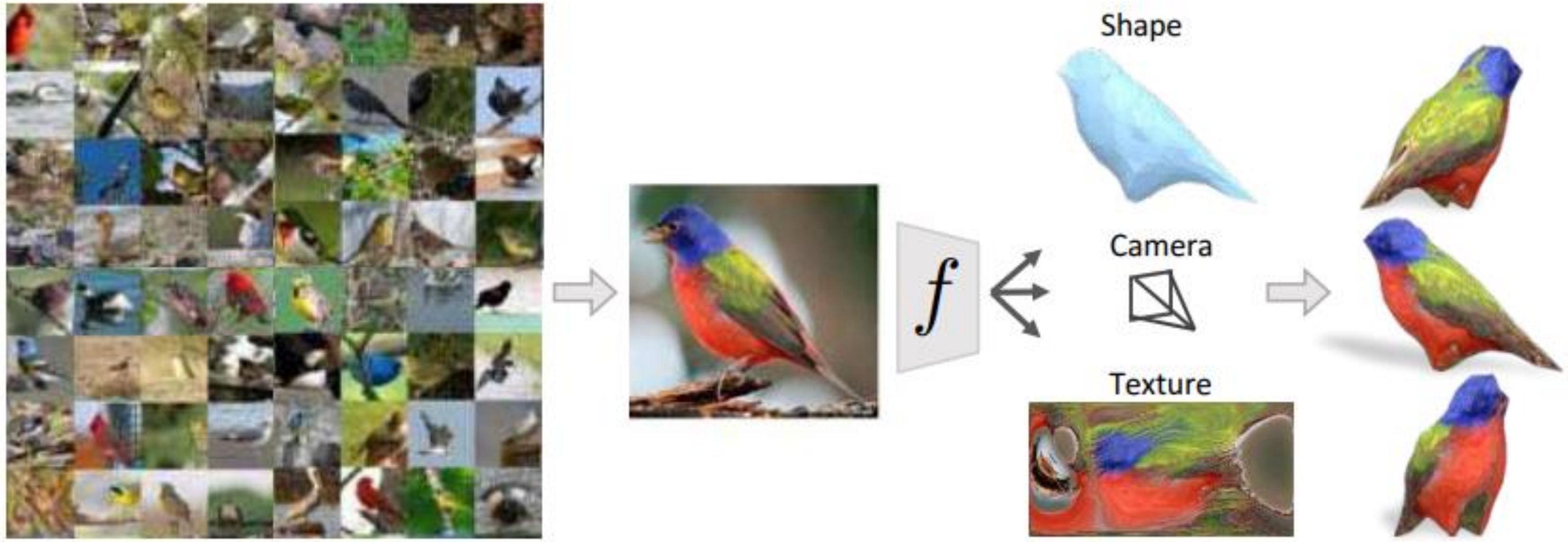
Propagation results

[1] Q. Zhou et al. Adaptive temporal encoding network for video instance-level human parsing. *arXiv preprint arXiv:1808.00661*, 2018

Application:
Dynamic Mesh Reconstruction from Videos in the Wild

Background

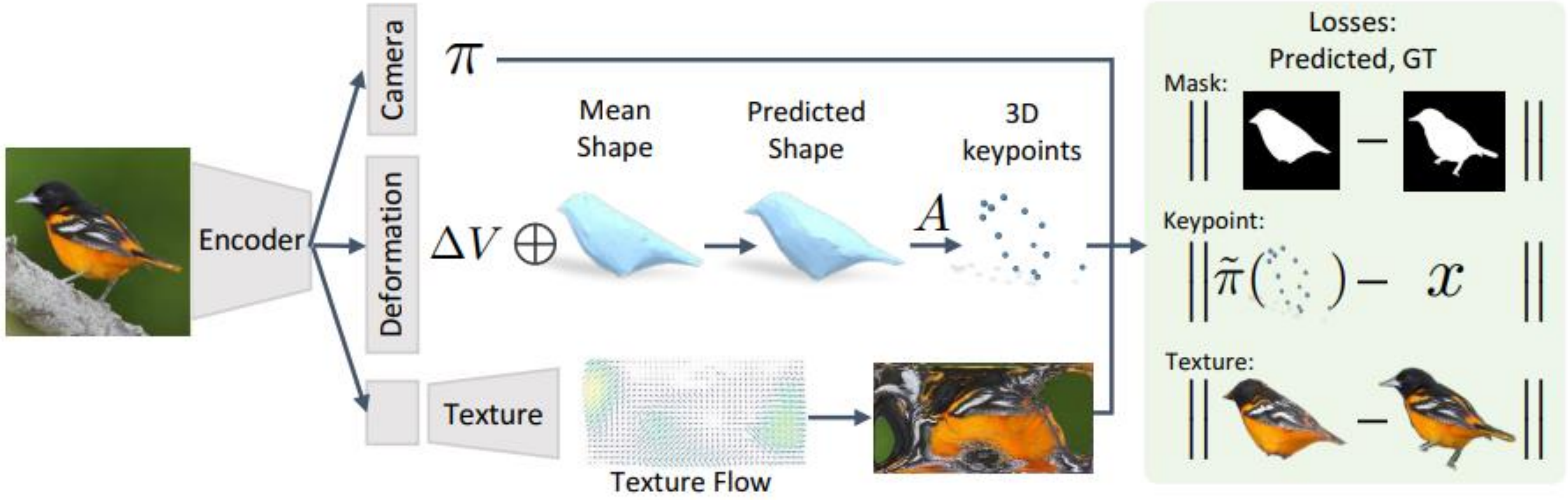
Reconstruct Object from an Image



Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, Jitendra Malik. Category-Specific Mesh Reconstruction. ECCV 2018

Background

Reconstruct Object from an Image



Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, Jitendra Malik. Category-Specific Mesh Reconstruction. ECCV 2018

Reconstruct Object from a Video

- Frame-wise applying the image model …

test video

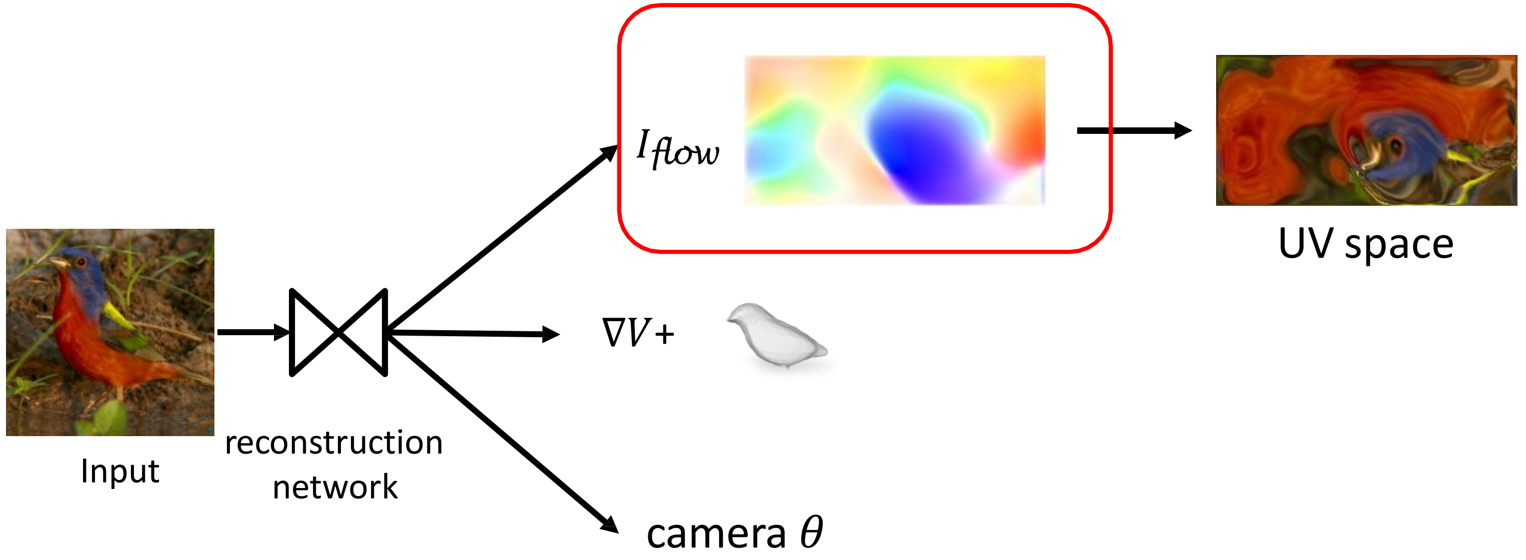
reconstructed

Reconstruct Object from a Video

- This is caused by:
  - low-quality video frames
  - small objects
  - appearance variations (lighting, clutter background, etc.)
  - domain gap
  - hard to annotate frame-wisely
  - …

Background

Reconstruct Object from an Image



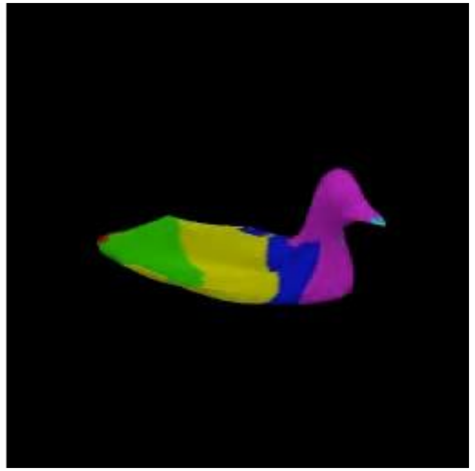
Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, Jitendra Malik. Category-Specific Mesh Reconstruction. ECCV 2018

Our Solution – Online Adaptation

For a single test video with only one instance, UV space will never change with the shape deformation and the camera translation.

test video

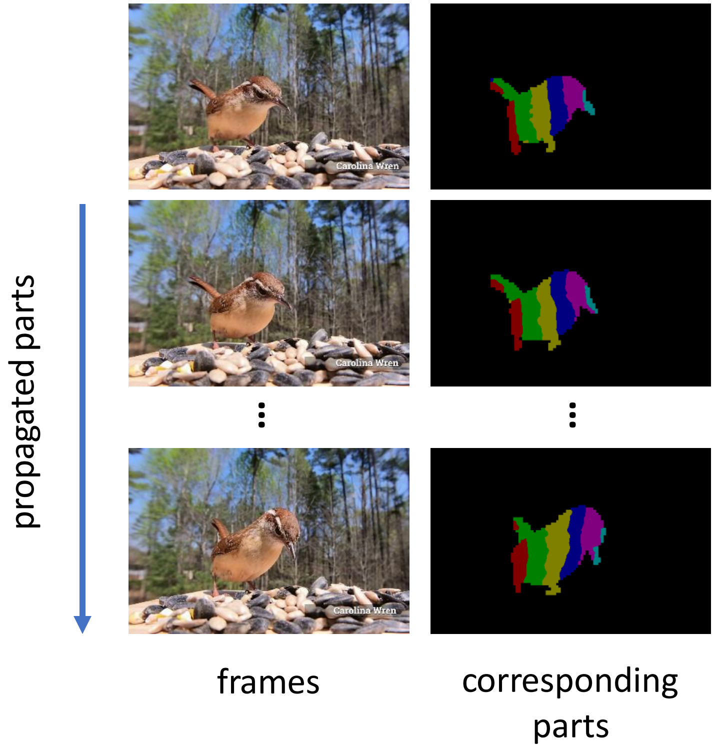
RGB texture



random parts

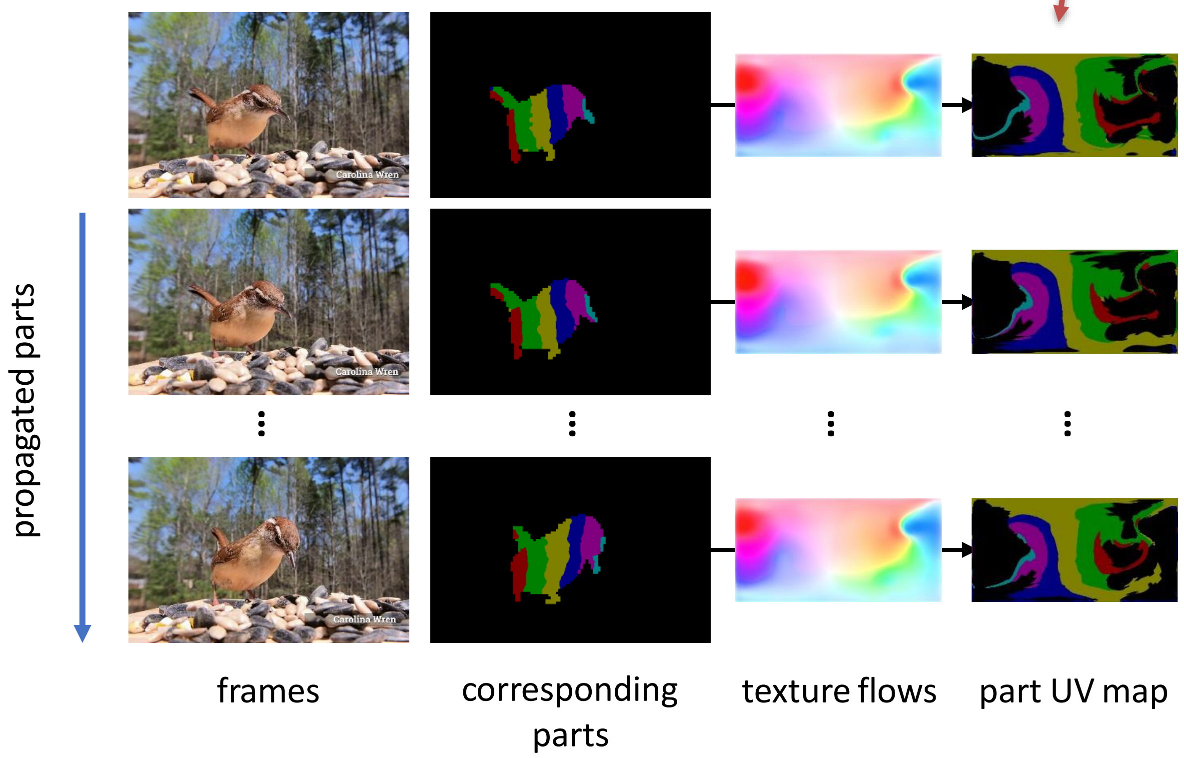
Part correspondence constraint

Randomly generate parts on the first frame



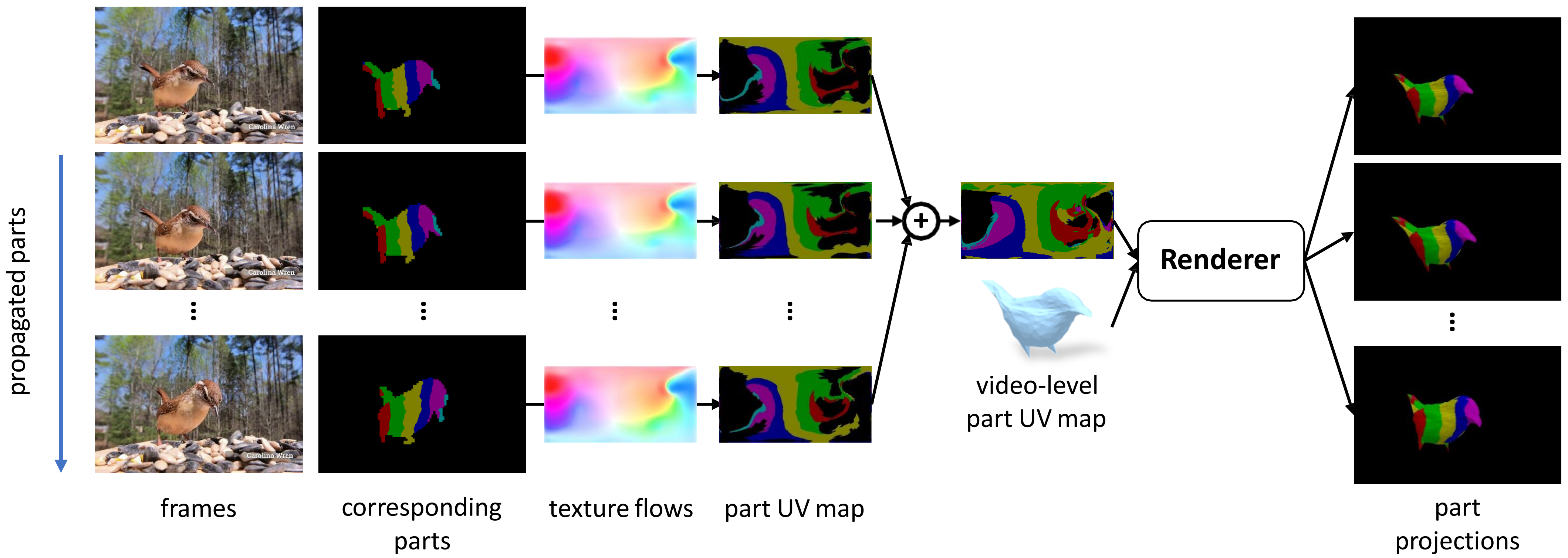
Part correspondence constraint

Map parts from each frame to the UV space by texture flows

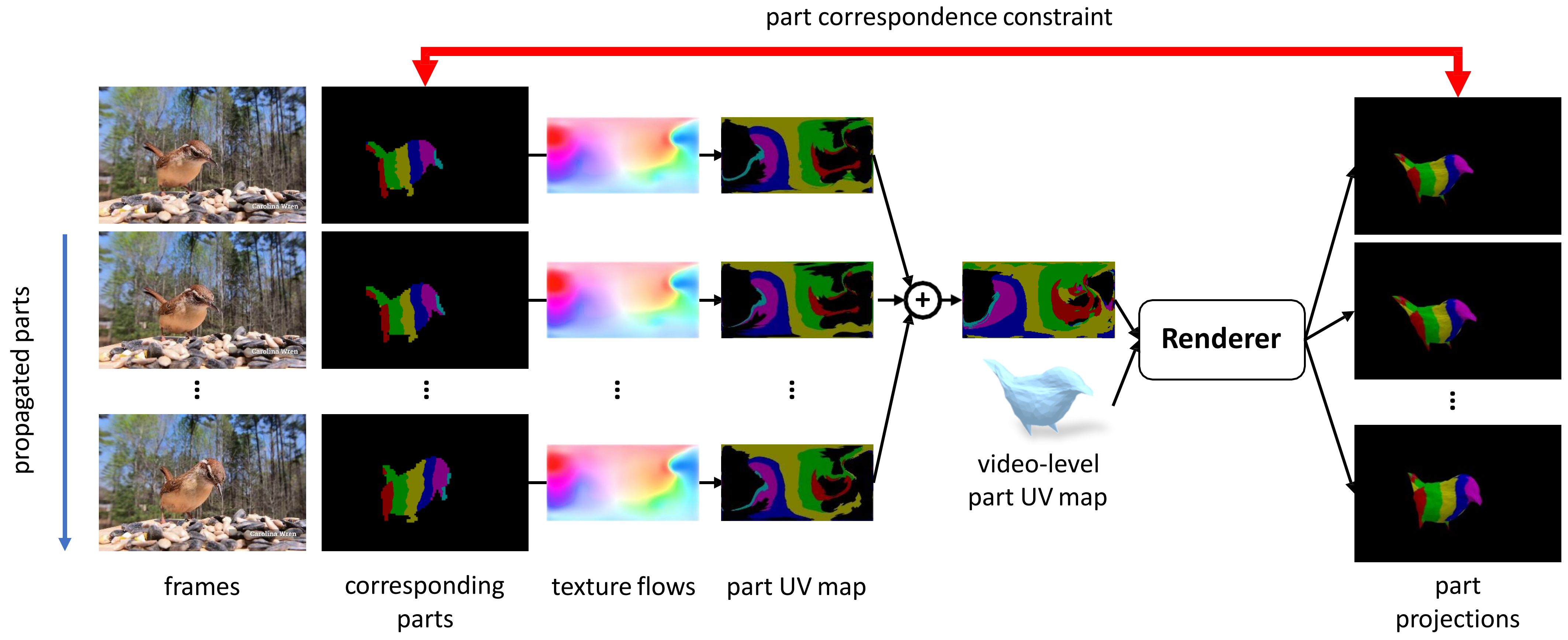


Part correspondence constraint

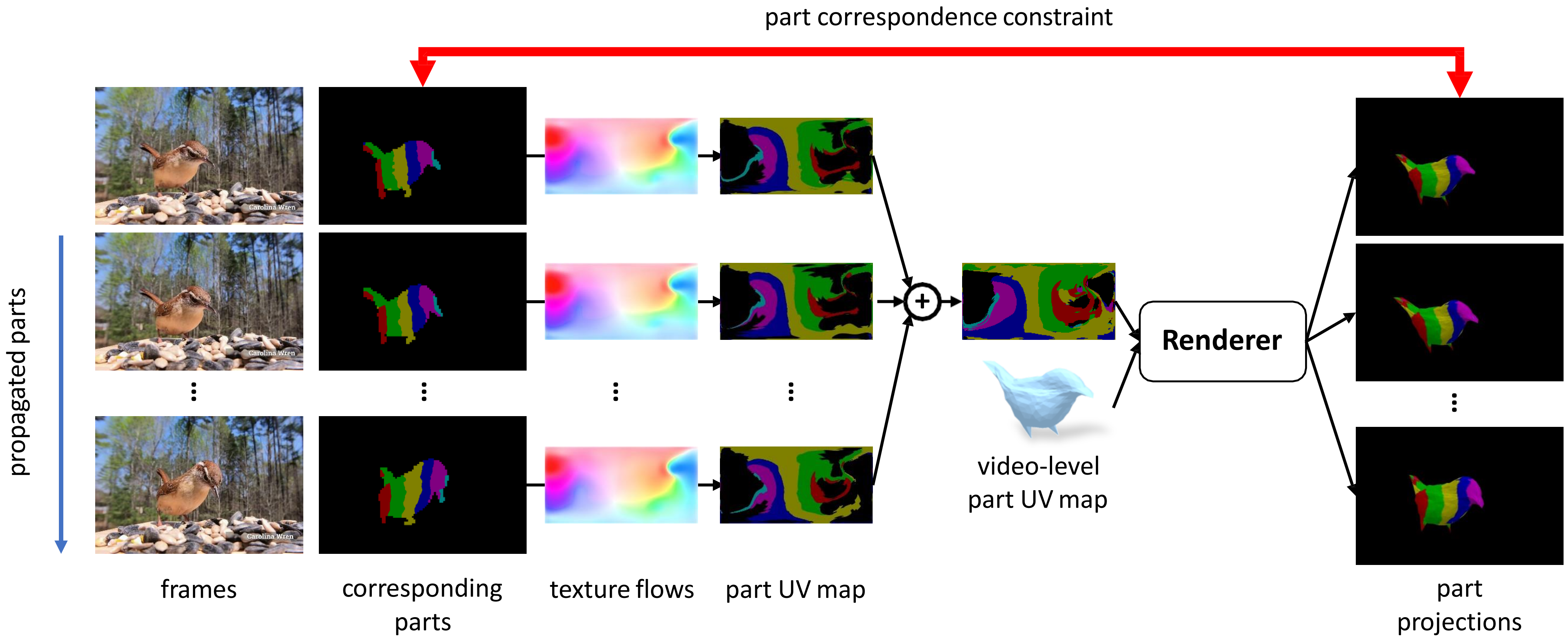
Wrap the video-level part UV map onto the base shape of each frame and render using predicted camera pose



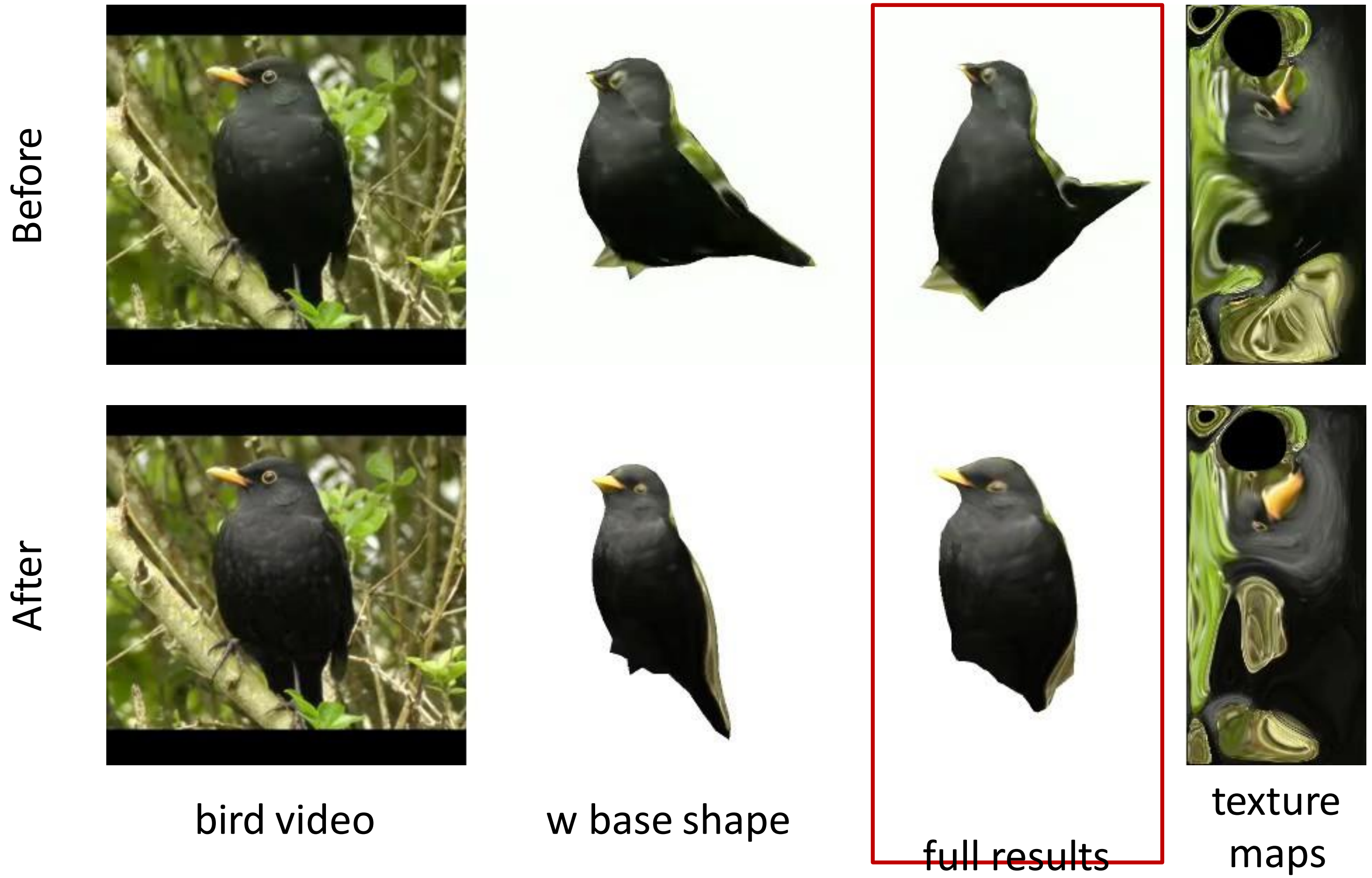
Part correspondence constraint



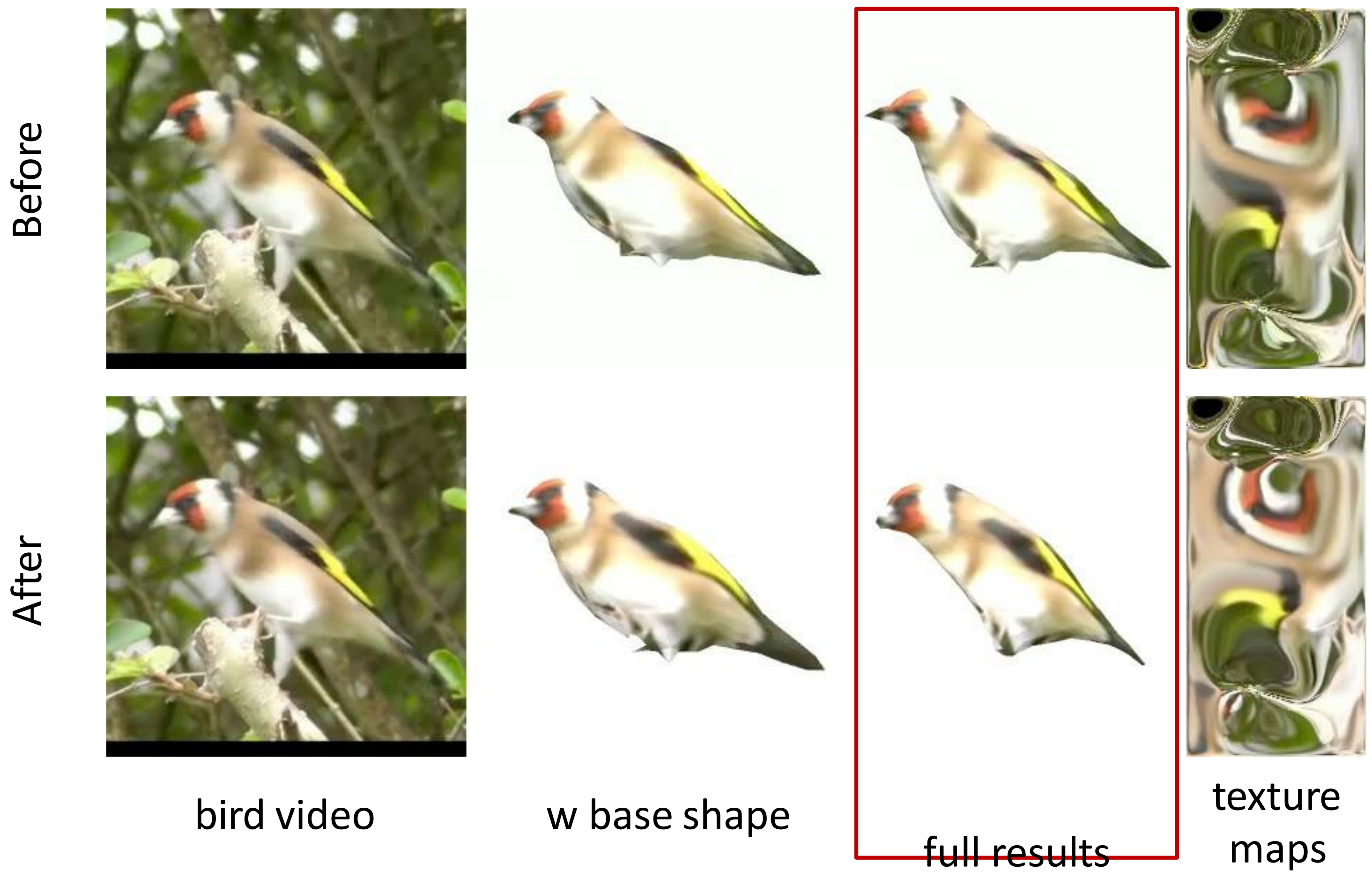
Part correspondence constraint



Video Reconstruction Results



Video Reconstruction Results



Conclusion

- KEY – Learning the inter-frame affinity matrix, which simultaneously models transitions between video frames at both the region- and pixel-levels.

Applications

Semi-supervised

Link different videos