Self-Supervised Cross-View Correspondence with Predictive Cycle Consistency

Supplementary Material

6. Implementation Details

In this section, we detail the specific setups for the following components:

1. Our Grayscale Colorization model (Section 3.2),
2. Our ViT Correspondence models (Section 3.5),
3. Our implementation of predictive approaches SiamMAE [18] and CroCoV2 [63], and
4. Our parameterization of SAM (Segment Anything Model) for tracking.

6.1. Hyperparameters

6.1.1 Grayscale and Correspondence Model

We implement our Grayscale Colorization model and Correspondence Model using the CroCoV2 [63] base architecture. Starting from the CroCoV2 Base-Decoder checkpoint, we continue pretraining with either the Grayscale Colorization objective or the original Cross-View MAE objective from CroCoV2 on datasets such as EgoExo4D [16] or Kinetics-400 [32]. The hyperparameters for continued pretraining are listed in Table 5.

Table 5. Hyperparameters for Grayscale Colorization and Correspondence Model Continual Training.

	Grayscale Colorization (Sec. 3.2)	Correspondence Model (Sec. 3.5)
Encoder Layers	12	12
Encoder Embed Dim	768	768
Decoder Layers	12	12
Decoder Embed Dim	768	768
MLP Dim	3072	3072
Learning rate	1.5×10^{-4}	1.5×10^{-4}
Adam β_1 / β_2	0.9 / 0.98	0.9 / 0.98
Weight decay	0.01	0.01
Learning rate schedule	Linear Decay	Linear Decay
Dropout	0.1	0.1
Warmup updates	8,000	8,000
Batch size	256	256
Updates	60,000	60,000
Training Objective	Colorization (RBG MSE Loss)	MAE
Kinetics-400 Time Gap	4-48 Frames	4-48 Frames

This continued pretraining results in the Grayscale Colorization model that we use to initialize PCC, extracting correspondence with the technique in Sec. 3.3. For our final PCC Correspondence Model, we further train using PCC pseudolabels as described in Section 3.5. Table 6 outlines the hyperparameters used for this additional training.

6.1.2 Baseline Implementation

To ensure fairness during evaluation, we continually pretrain CroCoV2 [63] and SiamMAE [18] on EgoExo4D [16] before measuring correspondence. The settings for continually pretraining CroCoV2 are outlined in Section 6.1.

Table 6. PCC Correspondence Model Hyperparameters. For each domain (EgoExo4D or Kinetics-400) we initialize our PCC Correspondence Model parameters with a continually pretrained MAE (Table 5)

	EgoExo4D [16] Correspondence	Kinetics-400 [16] Correspondence
Encoder Layers	12	12
Encoder Embed Dim	768	768
Decoder Layers	12	12
Decoder Embed Dim	768	768
MLP Dim	3072	3072
Learning rate	1.5×10^{-4}	1.5×10^{-4}
Adam β_1 / β_2	0.9 / 0.98	0.9 / 0.98
Weight decay	0.01	0.01
Learning rate schedule	Linear Decay	Linear Decay
Dropout	0.1	0.1
Warmup updates	2,000	2,000
Batch size	256	256
Updates	10,000	10,000
Training Objective	DICE + BCE	DICE + BCE
Kinetics-400 Time Gap	-	60 Frame Gap (2 sec)
EgoExo Parameters	50/50 Ego→Exo/Exo→Ego	-
Image size	240x240 (Ego) 240x416 (Exo)	224x224

Since the SiamMAE [18] code and checkpoints are not publicly available, we reimplement their approach by adapting the published CAT-MAE [30] codebase and checkpoints. We continually train SiamMAE using the CAT-MAE hyperparameters on Kinetics-400 for 60,000 steps with a batch size of 256. To validate our reimplementation, we evaluate our model on the DAVIS-2017 validation set [39], achieving a $\mathcal{J}\&\mathcal{F}_m$ score of 70.6, closely matching the original SiamMAE score of 71.4. For EgoExo4D, we continually pretrain this checkpoint for an additional 60000 steps at a batch size of 256, otherwise using the same settings.

We exclude DINO [37] models from continual pretraining on EgoExo4D due to a lack of diversity of data for image augmentation (EgoExo4D only has 123 unique sites used for data collection). Additionally, models employing exponentially moving average teachers require extensive tuning of the moving average temperature, making continual pretraining more challenging.

For our baselines, we adapt the K-Nearest-Neighbor implementation from [58]. While originally designed for multiple video frames, we modify it to treat all evaluation scenarios as two-frame videos. The algorithm inherently supports different resolutions for the first frame query and subsequent frames, accommodating the differing aspect ratios of Ego-view and Exo-view images. As detailed in Section 4.2, we resample all Ego and Exo videos to have a minimum resolution of 480p and perform a grid search to optimize the parameter k and the temperature. For EgoExo4D evaluation, we omit the neighborhood size parameter, as there is no spatial continuity between Ego and Exo views.

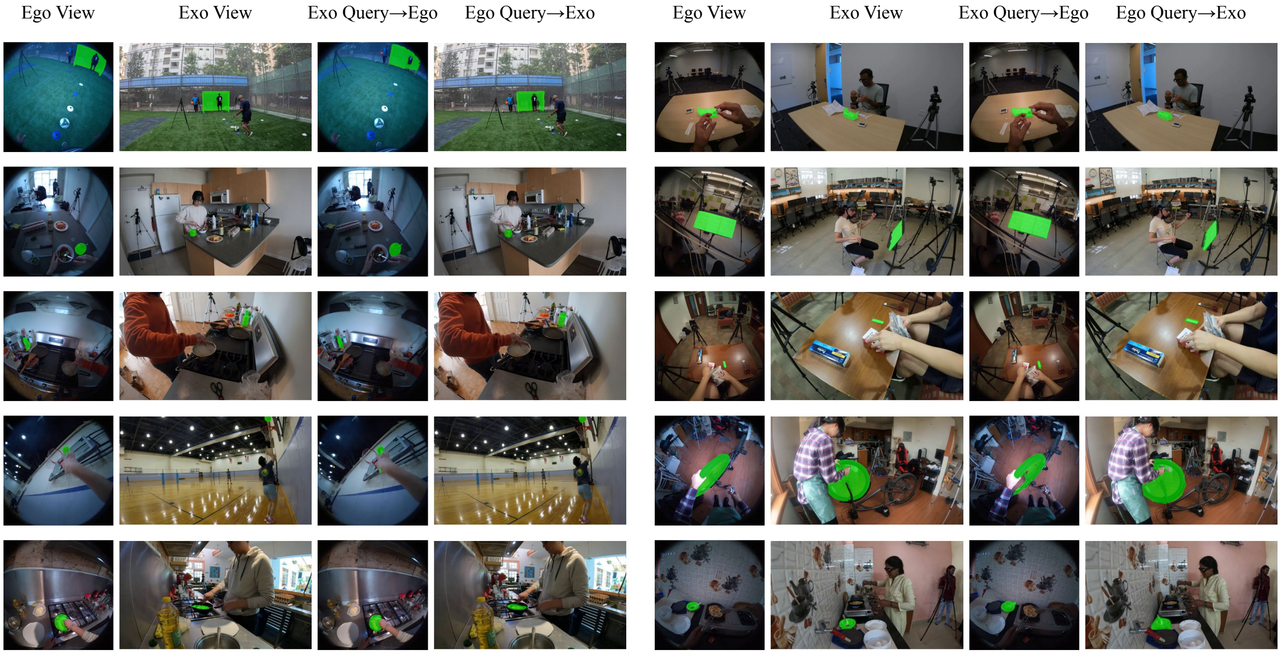

Figure 9. Additional Qualitative Results on the EgoExo4D Correspondence Benchmark.

However, for DAVIS-17 and LVOS evaluations, we independently grid search the neighborhood size parameter for each temporal distance.

We additionally compare against the state-of-the-art dense correspondence approach Probabilistic Warp Consistency [54] in Table 2. To implement this, we query for each pixel in the target view where it corresponds in the source view. Then, we say a pixel in the target view corresponds to a query object mask if its corresponding point in the source view is within the query mask. We choose the checkpoint trained with weak supervision on PF-Pascal [19], as this most closely matches the DAVIS-17 distribution for training.

6.1.3 SAM Configuration

To extract image segmentations from raw images in EgoExo4D, we use SAM with the standard point-grid prompting configuration, as demonstrated in [7, 34]. We note that this is different from the configuration of MASA [35], which uses bounding boxes extracted from an off-the-shelf object detection model using textual object descriptions. Because SAM is traditionally run on third-person videos, we gridsearch the Predicted IoU Threshold (0.88) and the Stability Score Threshold to (0.94) to have the highest IoU with ground truth object segmentation masks from the EgoExo4D validation set.

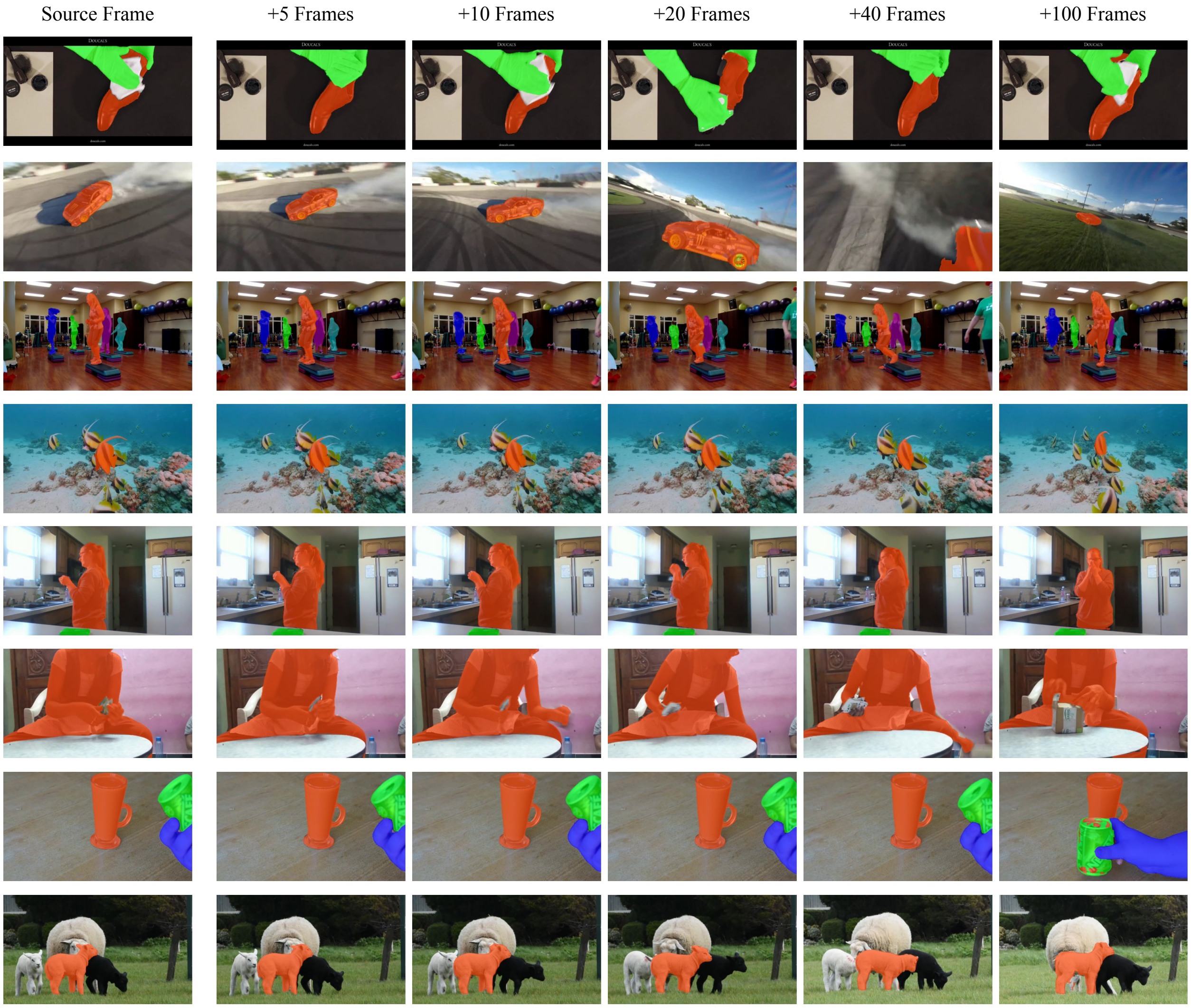

Figure 10. Additional Qualitative Results on LVOS with various frame gaps.