Supplementary Material: Procedure Learning via Regularized Gromov-Wasserstein Optimal Transport

Syed Ahmed Mahmood[†] Ali Shah Ali[†] Umer Ahmed[†] Fawad Javed Fateh[†]
M. Zeeshan Zia Quoc-Huy Tran

Retrocausal, Inc., Redmond, WA
www.retrocausal.ai

In this supplementary material, we first provide hyperparameter settings of our RGWOT approach across all datasets in Sec. S1. Next, we present sensitivity analyses of the margin λ and weight β of the C-IDM regularization, as well as the Gromov-Wasserstein weight α, structural prior radius r, and temporal prior weight ρ in Sec. S2, followed by run time comparisons of our RGWOT approach and competing methods including VAOT [1] and OPEL [3] in Sec. S3. We further provide an empirical justification of degenerate solutions in Sec. S4. Finally, we present quantitative results on all subtasks of egocentric and third-person datasets in Sec. S5.

S1. Hyperparameter Settings

Implementation details have been provided in Sec. 4 of the main paper. Here, we additionally provide hyperparameter settings of our RGWOT approach across all datasets. Tab. S1 lists hyperparameter settings for RGWOT, including the learning rate, optimizer, window size, and other relevant hyperparameters. Note that we use the same hyperparameter settings across all datasets.

Table S1. Hyperparameter settings for RGWOT.

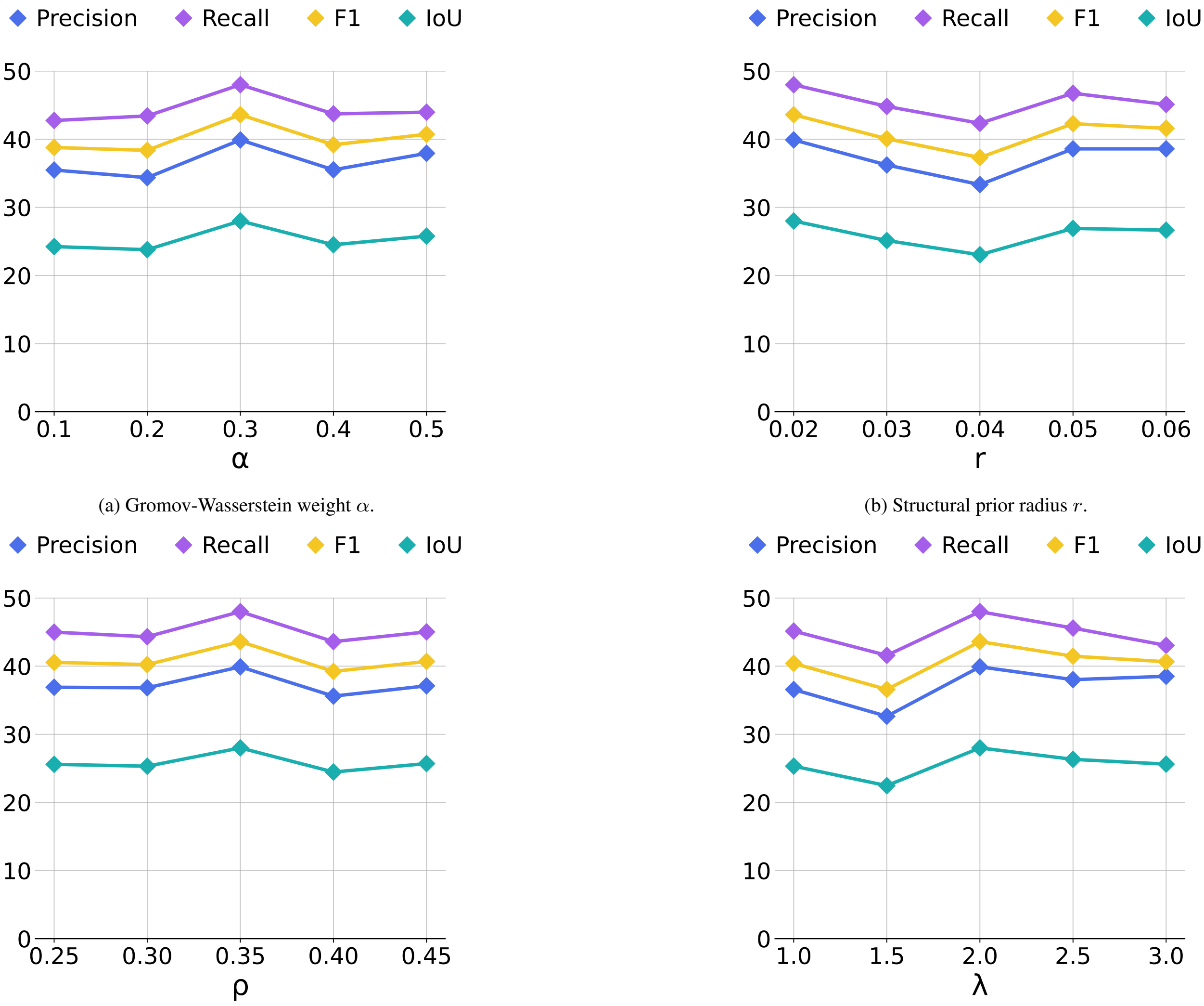
Hyperparameter	Value
No. of key-steps (K)	7
No. of sampled frames (X,Y)	32
No. of epochs	10000
Batch Size	2
Learning Rate (θ)	10^{-4}
Weight Decay	10^{-5}
Window size (σ)	300
Margin (λ)	2.0
No. of context frames (c)	2
Context stride	15
Embedding Dimension (D)	128
Optimizer	Adam [7]
Regularization parameter (ξ)	1.0
Entropy regularization weight (ϵ)	0.07
Virtual frame threshold (ζ)	0.5
Gromov-Wasserstein weight (α)	0.3
Structural prior radius (r)	0.02
Temporal prior weight (ρ)	0.35

S2. Sensitivity Analyses

In addition to the sensitivity analysis of K in Sec. 4.2 of the main paper, we further conduct sensitivity analyses on some key hyperparameters of our RGWOT approach, including the Gromov–Wasserstein weight α, structural prior radius r, temporal prior weight ρ, as well as the margin λ and weight β of the C-IDM regularization. First of all, Fig. S1 presents sensitivity analyses of the Gromov–Wasserstein weight α, structural prior radius r, temporal prior weight ρ, and margin λ on the EgoProceL [2] dataset (i.e., PC Assembly). Each subfigure illustrates how the Precision, Recall, F1, and IoU metrics change as the corresponding hyperparameter is adjusted. Overall, RGWOT demonstrates mostly stable performance across the studied ranges, with the performance metrics reaching their peaks at $\alpha = 0.3$, $r = 0.02$, $\rho = 0.35$, and $\lambda = 2$.

Next, we study the effects of the C-IDM regularization by varying its weight β in Eq. 7 of the main paper and include the results on the ProceL [5] dataset (i.e., make_smoke_salmon_sandwich) in Tab. S2. It is evident from the results that when β is small (e.g., $\beta = 0.01$), the learning collapses and RGWOT yields degenerate solutions, where all frames are mapped to a small cluster in the embedding space, and hence an entire video is assigned to a single key steps (see Fig. 3 for examples). Furthermore, when β increases (e.g., $\beta = 0.05$), the above issue is prevented.

[†] indicates joint first author.
{ahmed,alishah,umer,fawad,zeeshan,huy}@retrocausal.ai.



(a) Gromov-Wasserstein weight α.

(b) Structural prior radius r.

(c) Temporal prior weight ρ.

(d) Contrastive regularization margin λ.

Figure S1. Sensitivity analyses of Gromov-Wasserstein weight α, structural prior radius r, temporal prior weight ρ, and contrastive regularization margin λ.

β	**Precision**	**Recall**	**F1**	**IoU**
0.01	X	X	X	X
0.05	38.51	40.34	39.40	24.02
0.1	40.49	42.68	41.56	26.73
1	**42.61**	44.73	**43.64**	**28.22**
10	36.41	**44.87**	40.20	25.33
50	35.41	40.97	37.99	23.54

Table S2. Sensitivity analysis of the contrastive regularization weight β. Best results are in **bold**, while second best are underlined. 'X' denotes degenerate results.

Finally, the best performance is achieved when $\beta = 1$, indicating balancing weights for the video alignment loss and the contrastive regularization.

S3. Run Time Comparisons

In this section, we evaluate the efficiency (in terms of training times) of our RGWOT approach and competing methods including OPEL [3] and VAOT [1]. In particular, we train all methods on the EgoProceL [2] dataset (i.e., PC Disassembly) under identical experimental conditions (e.g., with 10,000 training epochs using two NVIDIA RTX 5090 GPUs). Tab. S3 presents the results. Thanks to having few losses and regularizers, our RGWOT approach is (notably) more efficient than OPEL [3], i.e., for training, OPEL [3] needs 206 minutes vs. 161 minutes for RGWOT. Thus, our approach is not only more accurate but also more efficient than OPEL [3]. In addition, RGWOT has similar training time as VAOT [1], i.e., for training, VAOT [1] re-

Table S3. Run time comparisons.

Method	Training Time
OPEL [3]	206 (mins)
VAOT [1]	161 (mins)
RGWOT (Ours)	164 (mins)

quires 164 minutes vs. 161 minutes for RGWOT, suggesting that adding contrastive regularization yields little extra computation.

S4. Degenerate Solution Justification

As discussed in Sec. 3.1.2 of the main paper, the objective in Eq. 5 minimizes visual and temporal differences between corresponding frames in X and Y only, and thus there is no mechanism to prevent the optimization from collapsing. Degenerate solutions appear consistently on the ProceL [5] dataset but not on the EgoProceL [2] and CrossTask [10] datasets. It is likely because 1) ADAM is a local optimizer, and 2) datasets exhibit different characteristics. To verify the latter, we extract ResNet [6] features (which are input to the model) from four EgoProceL videos and four ProceL videos and visualize them using t-SNE [9] in Figs. S2 and S3 respectively. We observe that EgoProceL features are more spread out, whereas ProceL features are more concentrated and hence easier to collapse. We believe this empirical evidence provides a convincing explanation and consider a theoretical analysis an interesting avenue for our future work.

S5. Quantitative Results on All Subtasks of Egocentric and Third-Person Datasets

Tabs. S4 and S5 present quantitative results on all subtasks of egocentric datasets, namely EGTEA-GAZE+ [8] and CMU-MMAC [4] respectively. Corresponding results for third-person datasets, including ProceL [5] and CrossTask [10], are provided in Tabs. S6 and S7 respectively. Our detailed evaluations span a wide range of scenarios, offering detailed assessments of our model performance from different viewpoints. The results highlight the robustness and versatility of our approach in addressing diverse videos and tasks, contributing to progress in procedure learning and related fields.

References

[1] Ali Shah Ali, Syed Ahmed Mahmood, Mubin Saeed, Andrey Konin, M Zeeshan Zia, and Quoc-Huy Tran. Joint self-supervised video alignment and action segmentation. *arXiv preprint arXiv:2503.16832*, 2025. 1, 2, 3

[2] Siddhant Bansal, Chetan Arora, and CV Jawahar. My view is the best view: Procedure learning from egocentric videos. In *European Conference on Computer Vision*, pages 657–675. Springer, 2022. 1, 2, 3, 4

[3] Sayeed Shafayet Chowdhury, Soumyadeep Chandra, and Kaushik Roy. Opel: Optimal transport guided procedure learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 1, 2, 3

[4] Fernando De la Torre, Jessica Hodgins, Adam Bargteil, Xavier Martin, Justin Macey, Alex Collado, and Pep Beltran. Guide to the carnegie mellon university multimodal activity (cmu-mmac) database. 2009. 3, 5

[5] Ehsan Elhamifar and Dat Huynh. Self-supervised multi-task procedure learning from instructional videos. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pages 557–573. Springer, 2020. 1, 3, 4, 5

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[8] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European conference on computer vision (ECCV)*, pages 619–635, 2018. 3, 5

[9] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (Nov):2579–2605, 2008. 3, 4

[10] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545, 2019. 3, 5

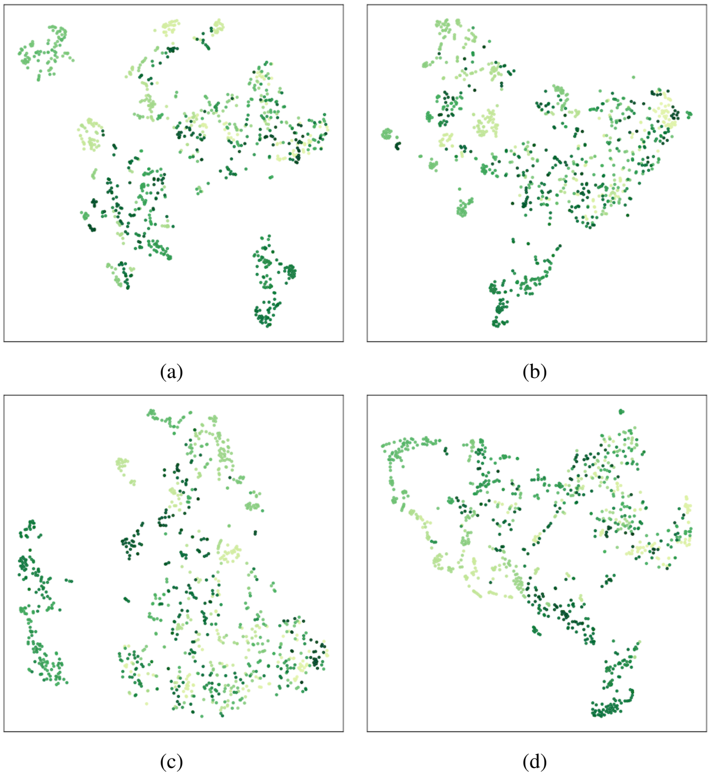


Figure S2. t-SNE [9] plots of the EgoProceL [2] dataset. Frame order from first to last is indicated by a gradient from light to dark green.

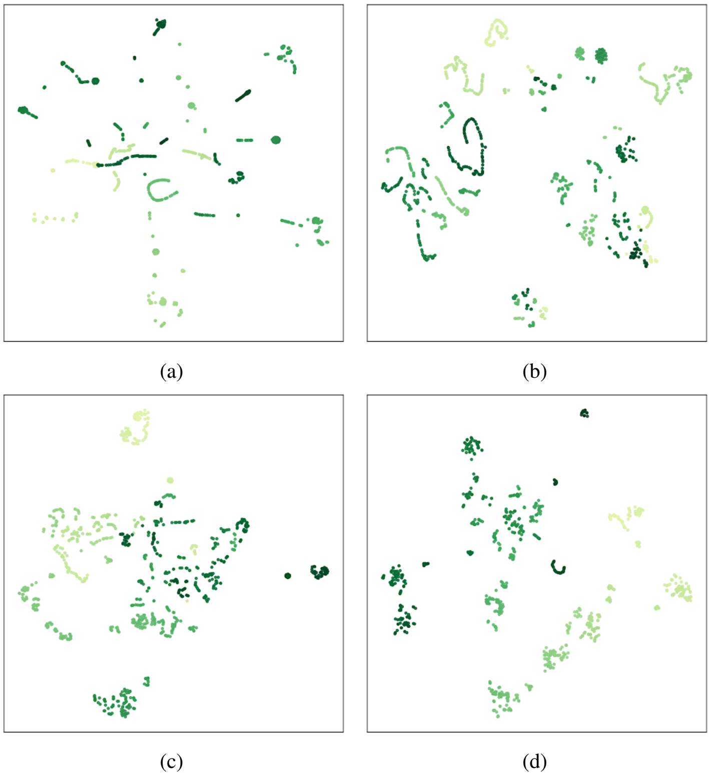


Figure S3. t-SNE [9] plots of the ProceL [5] dataset. Frame order from first to last is indicated by a gradient from light to dark green.

Table S4. Results on individual subtasks of the EGTEA-GAZE+ [8] dataset.

	Bacon Eggs	Cheeseburger	Breakfast	Greek Salad	Pasta Salad	Pizza	Turkey
F1	37.8	41.2	34.5	37.3	35.0	37.1	38.7
IoU	23.4	26.2	21.0	23.0	21.4	22.9	22.8

Table S5. Results on individual subtasks of the CMU-MMAC [4] dataset.

	Brownie	Eggs	Sandwich	Salad	Pizza
F1	59.5	44.1	56.1	68.6	43.7
IoU	42.8	28.5	39.9	53.3	28.2

Table S6. Results on individual subtasks of the ProceL [5] dataset.

	Assemble Clarinet	Change iPhone Battery	Change Tire	Change Toilet Seat	Jump Car	Coffee
F1	56.4	48.8	42.7	48.2	35.2	48.8
IoU	41.6	33.5	27.6	32.5	21.7	33.3
	Make PBJ Sandwich	Make Salmon Sandwich	Perform CPR	Repot Plant	Setup Chromecast	Tie Tie
F1	38.8	43.6	42.1	47.4	36.6	43.7
IoU	24.3	28.2	27.1	32.2	22.7	28.1

Table S7. Results on individual subtasks of the CrossTask [10] dataset.

	40567	16815	23521	44047	44789	77721	87706	71781	94276
F1	42.8	53.8	33.9	34.7	31.7	38.0	33.1	32.8	39.5
IoU	27.2	37.4	20.6	21.2	18.9	23.7	20.3	19.8	25.1
	53193	76400	91515	59684	95603	105253	105222	109972	113766
F1	38.5	34.4	52.3	36.1	53.8	52.0	40.4	42.8	37.4
IoU	24.1	20.9	37.1	23.6	38.4	37.6	26.3	28.0	23.4