Every Shot Counts: Using Exemplars for Repetition Counting in Videos – Appendix

Code is made publicly available at: https://github.com/sinhasaptarshi/EveryShotCounts. The repository contains the full train and evaluation code and a demo for inference with a few videos.

In the following sections, we provide more qualitative results in Sec. 6. We then provide additional ablations on the architecture's choices (e.g. depth of transformer and window size) in Sec. 7. Additionally, we evaluate the ability of ESCounts to locate each repetition within the video in Sec. 8. We then compare VRC to Temporal Action Segmentation (TAS) in Sec. 9 demonstrating distinctions between the two tasks.

Additionally, following the release of the recent egocentric video counting dataset OVR-Ego4D [3], we train and evaluate ESCounts on this newly introduced dataset demonstrating the effectiveness of our method for egocentric counting in Sec. 10.

Table 5: Impact of L.

L	RMSE↓	MAE↓	OBZ↑	OBO↑
1	4.843	0.229	0.223	0.545
2	**4.455**	**0.213**	0.245	**0.563**
3	4.575	0.219	**0.247**	0.560
4	4.783	0.225	0.235	0.548

Table 6: Impact of L'.

L'	RMSE↓	MAE↓	OBZ↑	OBO↑
1	4.932	0.247	0.212	0.525
2	4.634	0.218	0.238	0.550
3	**4.455**	**0.213**	**0.245**	**0.563**
4	4.532	0.225	0.230	0.552

Table 7: Window sizes.

(t', h', w')	RMSE↓	MAE↓	OBZ↑	OBO↑
$(3, 3, 3)$	5.212	0.261	0.185	0.521
$(2, 7, 7)$	4.871	0.247	0.201	0.537
$(4, 7, 7)$	**4.455**	**0.213**	**0.245**	**0.563**
$(7, 7, 7)$	4.753	0.225	0.232	0.520
full	5.011	0.227	0.221	0.533

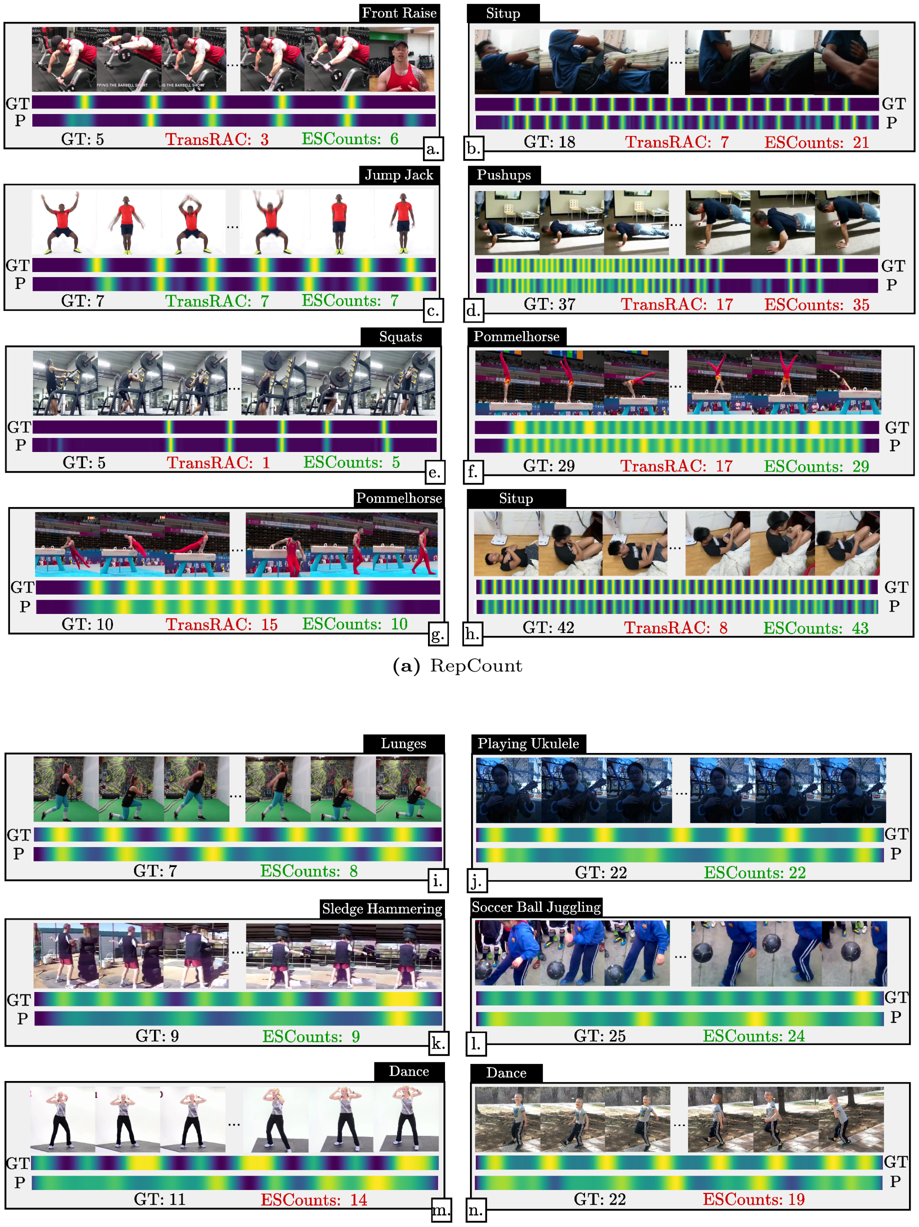
6 Qualitative Video and Extended Figure

We provide a compilation of videos on our website https://sinhasaptarshi.github.io/escounts/ showcasing our method's Video Repetition Counting (VRC) abilities over a diverse set of 20 videos from all 3 datasets. Videos are shown alongside synchronised ground truth and predicted density maps. The test set from which each video is sampled is also shown.

We additionally extend Fig. 5 in the main paper with more examples from all datasets in Fig. 7.

7 Further Ablations

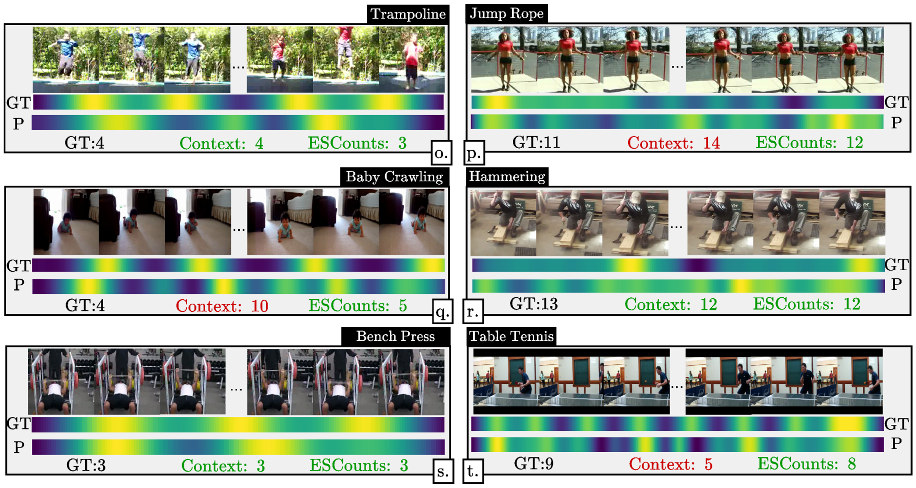
We extend the ablations in Sec. 4.3, report results over different L and L', and analyse the impact of windowed-self attention on the performance of ESCounts.



(a) RepCount

(b) Countix

Fig. 7: Additional qualitative results.



(c) UCFRep

Fig. 7: Additional qualitative results (continued).

Impact of L. We ablate L *i.e.* the number of layers in the cross-attention block. Increasing L increases the number of operations that discover correspondences between the video and the selected exemplars. As seen in Tab. 5, while low L causes a drop in performance, high L can also be detrimental probably due to overfitting. $L = 2$ gives the best results for the majority of the metrics.

Next keeping $L = 2$ fixed, we vary L' in Tab. 6. L' is the number of windowed self-attention layers in the self-attention block. $L' = 3$ gives the best results across all the metrics. Similarly, increasing or decreasing L' drops performance gradually.

Self-attention vs Windowed Self-attention. Motivated by [9], we use windowed self-attention for the decoder self-attention blocks. Given spatio-temporal tokens $\mathcal{T}' \times H' \times W' \times C$, windowed self-attention computes multi-headed attention for each token within the immediate neighbourhood using 3D shifted windows of size $t' \times h' \times w'$, where $t' \leq \mathcal{T}'$, $h' \leq H'$ and $w' \leq W'$. We ablate on various (t', h', w') values in Tab. 7. Note that for $t' = \mathcal{T}'$, $h' = H'$, and $w' = W'$ denoted as *full*, standard self-attention is used where each token attends to every token. As shown, the best performance is obtained with window size $(4, 7, 7)$, demonstrating the importance of attending to tokens in immediate spatio-temporal neighbourhoods only. We found variations in the value of t' to have the largest performance impact with decreases as the value of t' changes.

Sampling Rate for Encoding. As stated in the implementation details, we sample every four frames from the video to form the encoder inputs. We ablate the impact of the sampling rate in Tab. 8. As shown, denser sampling is key for robust video repetition counting. Reducing the sampling rate steadily decreases performance as relevant parts of repetitions may be missed.

Table 8: Impact of sampling rate

Sampling every n frames	RMSE ↓	MAE ↓	OBZ↑	OBO↑
4	**4.455**	**0.213**	**0.245**	**0.563**
8	5.112	0.268	0.221	0.521
16	5.911	0.296	0.185	0.482
32	6.562	0.346	0.156	0.444

Table 9: OBO, parameters, and training and inference speeds on UCFRep. Metrics obtained by the public available codebase of [11] are denoted with *.

Method	OBO↑	#Trainable params (M)	Train set ↓ (sec/sample)	Test set ↓ (sec/sample)
Context [11]	**0.790**	47.6*	1.171*	1.818*
ESCounts	0.731	**21.1** (-26.5)	**0.138** (-1.033)	**0.141** (-1.677)

Model Size and Speed For UCFRep [11], [11, 12] achieve better performance than ESCounts. However, this performance is achieved by having more trainable parameters, as [11, 12] finetune the encoders on the target dataset. We use the provided codebase from [11] and benchmark the average number of iterations per second for a full forward and backward pass over the entire training set. Additionally, we report inference-only average times on the test set. We use the same experiment set-up described in Sec. 4.1 and report speeds in Tab. 9. Training ESCounts is ∼8× faster. Interestingly, ESCounts maintains its efficiency even during inference with ∼12× faster times than Context [11] which uses iterative processing. Note that [12] could not be used for this analysis as their code for training with UCFRep is not publicly available.

8 Repetition Localisation

VRC metrics only relate predicted to correct counts, regardless of whether the repetitions have been correctly identified. We thus investigate whether the peaks of the predicted density map $\tilde{\mathbf{d}}$ align with the annotated start-end times of repetitions in the ground truth. Following action localisation methods [1, 5, 8], we adopt the Jaccard index $\mathcal{J}$ for repetition localisation. As the values of $\tilde{\mathbf{d}}$ peaks vary across videos, we apply thresholds θ relative to the maximum and minimum values, $r = \theta(\max(\tilde{\mathbf{d}}) - \min(\tilde{\mathbf{d}}))$. We find all local maxima in $\tilde{\mathbf{d}}$ and only keep

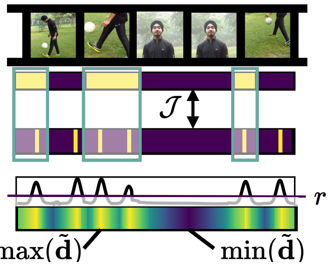


Fig. 8: Localisation metric $\mathcal{J}$. We identify local maxima in $\tilde{\mathbf{d}}$ and threshold peaks higher than r to remove noise. $\mathcal{J}$ is then computed between the annotated start-end times and the thresholded peaks.

Table 10: Repetition localisation results on RepCount measured as the mAP (%) over different Jaccard index relative thresholds r.

Method	θ values for relative threshold r									Avg
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	
Baseline [6]	38.59	37.46	35.02	32.55	30.40	26.97	22.66	17.22	12.17	28.12
ESCounts	**38.83**	**38.64**	**38.07**	**37.44**	**35.82**	**33.43**	**30.76**	**27.52**	**20.85**	**33.48**

those above threshold r. We consider a repetition to be correctly located (TP) if at least one peak occurs within the start-end time of that repetition. Peaks that occur within the same repetition are counted as one. In contrast, peaks that do not overlap with repetitions are false positives (FP) and repetitions that do not overlap with any peak are false negatives (FN). We then calculate $\mathcal{J}$ as TP divided by all the correspondences (TP + FP + FN) as customary.

In Tab. 10 we report the Jaccard index over different thresholds alongside the Mean Average Precision (mAP) on RepCount. We select TransRAC [6] as a baseline due to their publicly available checkpoint. Across thresholds, ESCounts outperforms [6] with the most notable improvements observed over higher threshold values. This demonstrates ESCounts' ability to predict density maps with higher contrast between higher and lower salient regions. For 0.9, 0.8, and 0.7 thresholds ESCounts demonstrates a +8.68%, +10.30%, and +8.10% improvement over [6].

Table 11: Comparison between ESCounts and TAS baseline on close and open-set RepCount setting.

Task	Method	benchmark		open-set	
		MAE↓	OBO↑	MAE↓	OBO↑
TAS	GTRM [7]	0.527	0.159	1.000	0.000
	TriDet [10]	0.603	0.232	1.000	0.000
VRC	ESCounts	**0.213**	**0.563**	**0.436**	**0.519**

9 Distinction between VRC and TAS

Unlike Temporal Action Segmentation (TAS) methods, VRC methods can generalise to unseen action classes. In Tab. 11 we compare ESCounts to a TAS method [7] on the RepCount benchmark (*close-set*) and *open-set* setting. As shown, [7] can only localise the actions of a pre-defined set of categories with which the model was trained. In contrast, VRC is learned as an *open-set* task. As ESCounts uses a learnt latent to encode class-independent repetition embeddings, it effectively generalises to unseen categories. In addition, ESCounts can

Table 12: Results on OVR-Ego4D.† indicates results have been copied from [3]. (V) corresponds to vision-only models and (V+L) to vision and language models.

Modality	Method	RMSE ↓	MAE ↓	OBZ↑	OBO↑
V	RepNet [2] †	3.20	0.74	0.19	0.43
	ESCounts	2.41	**0.32**	**0.30**	**0.68**
V+L	OVRCounter [3] †	**1.60**	0.35	0.29	0.66

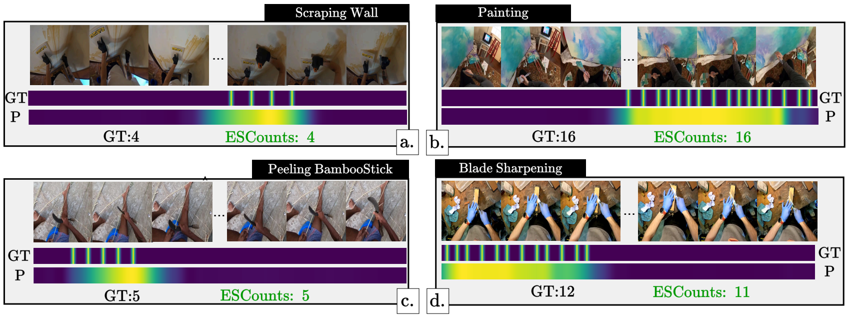


Fig. 9: Qualitative results of ESCounts on OVR-Ego4D. For the selected videos, we show both ground truth (GT) and predicted (P) density maps along with the counts. Note that for OVR-Ego4D, we do not have temporal annotations for individual repetitions. Therefore similar to Countix, we show pseudo-labels as the GT density maps.

better handle large variations in repetition durations that are present in VRC videos compared to [7], which as noted by [6] is a weakness of TAS methods.

10 Results on egocentric VRC

The recently-introduced OVR-Ego4D [3] is an Ego4D [4] subset containing clips of repetitive egocentric actions, *e.g.* cutting onions, rolling dough. It comprises 50.6K 10-second clips with 41.9K train and 8.7K test clips. Annotations are only provided for the number of repetitions and not the individual start and end times per repetition. Thus, similar to Countix, we define pseudo-labels to estimate the density maps. We evaluate ESCounts on OVR-Ego4D in Tab. 12. Compared to the vision-language-based OVRCounter, [3] ESCounts improves OBZ, OBO, and MAE, with only visual inputs, *without any language input in training or inference*, showing ESCounts' effectiveness for the domain of egocentric counting. We also add some qualitative results in Fig. 9. Similar to results on other datasets, ESCounts predicts accurate counts a over diverse range of counts. The peaks of individual repetitions are not as clear, due to the pseudo-labels, but ESCounts correctly finds the OBO counts in each case.

References

1. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 961–970 (2015)
2. Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., Zisserman, A.: Counting Out Time: Class Agnostic Video Repetition Counting in the Wild. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10387–10396 (2020)
3. Dwibedi, D., Aytar, Y., Tompson, J., Zisserman, A.: Ovr: A dataset for open vocabulary temporal repetition counting in videos. arXiv preprint arXiv:2407.17085 (2024)
4. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., Martin, M., Nagarajan, T., Radosavovic, I., Ramakrishnan, S.K., Ryan, F., Sharma, J., Wray, M., Xu, M., Xu, E.Z., Zhao, C., Bansal, S., Batra, D., Cartillier, V., Crane, S., Do, T., Doulaty, M., Erapalli, A., Feichtenhofer, C., Fragomeni, A., Fu, Q., Gebreselasie, A., González, C., Hillis, J., Huang, X., Huang, Y., Jia, W., Khoo, W., Kolář, J., Kottur, S., Kumar, A., Landini, F., Li, C., Li, Y., Li, Z., Mangalam, K., Modhugu, R., Munro, J., Murrell, T., Nishiyasu, T., Price, W., Ruiz, P., Ramazanova, M., Sari, L., Somasundaram, K., Southerland, A., Sugano, Y., Tao, R., Vo, M., Wang, Y., Wu, X., Yagi, T., Zhao, Z., Zhu, Y., Arbeláez, P., Crandall, D., Damen, D., Farinella, G.M., Fuegen, C., Ghanem, B., Ithapu, V.K., Jawahar, C.V., Joo, H., Kitani, K., Li, H., Newcombe, R., Oliva, A., Park, H.S., Rehg, J.M., Sato, Y., Shi, J., Shou, M.Z., Torralba, A., Torresani, L., Yan, M., Malik, J.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 18995–19012 (June 2022)
5. Gu, C., Sun, C., Ross, D.A., Vondrick, C., Pantofaru, C., Li, Y., Vijayanarasimhan, S., Toderici, G., Ricco, S., Sukthankar, R., et al.: AVA: A Video Dataset of Spatio-Temporally Localized Atomic Visual Actions. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6047–6056 (2018)
6. Hu, H., Dong, S., Zhao, Y., Lian, D., Li, Z., Gao, S.: TransRAC: Encoding Multi-scale Temporal Correlation with Transformers for Repetitive Action Counting. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19013–19022 (2022)
7. Huang, Y., Sugano, Y., Sato, Y.: Improving action segmentation via graph-based temporal reasoning. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14024–14034 (2020)
8. Idrees, H., Zamir, A.R., Jiang, Y.G., Gorban, A., Laptev, I., Sukthankar, R., Shah, M.: The thumos challenge on action recognition for videos “in the wild". Computer Vision and Image Understanding (CVIU) **155**, 1–23 (2017)
9. Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., Hu, H.: Video Swin Transformer. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3202–3211 (2022)
10. Shi, D., Zhong, Y., Cao, Q., Ma, L., Li, J., Tao, D.: Tridet: Temporal action detection with relative boundary modeling. In: CVPR (2023)
11. Zhang, H., Xu, X., Han, G., He, S.: Context-Aware and Scale-Insensitive Temporal Repetition Counting. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 670–678 (2020)
12. Zhang, Y., Shao, L., Snoek, C.G.: Repetitive Activity Counting by Sight and Sound. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14070–14079 (2021)