Online Detection of End of Take and Release Actions from Egocentric Videos

Alessandro Sebastiano Catinello, Giovanni Maria Farinella and Antonino Furnari
Department of Mathematics and Computer Science, University of Catania, Italy
ale.catinello.c@gmail.com, {giovanni.farinella, antonino.furnari}@unict.it

Keywords: Online Action Detection, Take/Release Action Detection, Egocentric Untrimmed Video Analysis.

Abstract: In this work, we tackle the problem of detecting "take" and "release" actions from egocentric videos. We address the task following a new Online Detection of Action End (ODAE) formulation in which algorithms have to determine the end of an action in an online fashion. We show that ODAE has advantages over previous formulations that focus on detecting actions at the contact frame or offline, thanks to the reduced uncertainty due to the complete observation of events before a prediction is made. We adapt to this task and benchmark different state-of-the-art temporal online action detection models on the EPIC-KITCHENS dataset, highlighting the specific challenges of the ODAE task, such as sparse annotations and high action density. Analysis on THUMOS14 shows that most conclusions are valid also in a third-person vision scenario. We also investigate the impact of techniques such as label propagation to address annotation imbalance. Our results show that the problem is far from being solved, Mamba-based models consistently outperform transformer-based models in all settings.

1 INTRODUCTION

Wearable devices observe the world from the user's perspective, enabling user-centric applications that assist in daily tasks (Plizzari et al., 2024). Understanding atomic actions such as "take" (picking up an object) and "release" (putting down an object) is essential for assistive systems, enabling applications like action anticipation, object usage tracking, or error detection during tasks.

Wearable devices observe the world from the user's perspective, enabling user-centric applications that assist in daily tasks (Plizzari et al., 2024). Understanding atomic actions such as "take" (picking up an object) and "release" (putting down an object) is essential for assistive systems, enabling applications like action anticipation, object usage tracking, or error detection during tasks. While related tasks, such as hand-object interaction detection (Shan et al., 2020; Darkhalil et al., 2022; Cheng et al., 2023), object-state change recognition (Grauman et al., 2022; Xue et al., 2024; Souček et al., 2022), temporal action detection (Zhang et al., 2022; Wang et al., 2021a; Wang et al., 2021b; Liu et al., 2024), and online action recognition (Chen et al., 2024; Zhao and Krähenbühl, 2022; Wang et al., 2021c; An et al., 2023), have been explored, none fully address the requirements for take/release detection. Specifically, this task should operate at the video level, in an online fashion, and ensure temporal consistency by signaling a single action per occurrence.

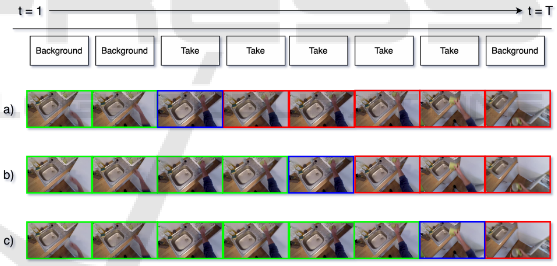

Figure 1: Different schemes for the detection of take/release actions from egocentric videos. Frames marked in blue denote the time at which models are requested to predict the ground truth take action when performing (a) detection of action start, (b) detection at contact frame, and (c) detection of action end. Predicting action end times (c) is less ambiguous than anticipating actions before observing them (a) or at the contact frame for partial observations (b). For instance in (b) it would be hard to distinguish a "touch" from a "take" action.

Inspired by the Online Detection of Action Start (ODAS) task (Shou et al., 2018), we formulate the detection of take/release actions as an "Online Detection of Action End (ODAE)" task, which focuses on identifying action completion in egocentric video

Catinello, A. S., Farinella, G. M. and Furnari, A.
Online Detection of End of Take and Release Actions from Egocentric Videos.
DOI: 10.5220/0013249700003912

In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 2: VISAPP*, pages 863-870
ISBN: 978-989-758-728-3; ISSN: 2184-4321

streams in real-time. ODAE aims to signal actions immediately after they conclude, avoiding ambiguities associated with incomplete observations or early predictions (Scavo et al., 2023). Methods are required to output one prediction per action, penalizing missed detections, multiple detections, and overly delayed or early predictions.

We provide an in-depth investigation of the ODAE task, benchmarking state-of-the-art temporal action detection models (Zhang et al., 2022; Zhao and Krähenbühl, 2022) on EPIC-KITCHENS-100 and THUMOS datasets in both online and offline settings. The study highlights the challenges posed by sparse ground truth annotations and high action density in egocentric scenarios. To address these, we evaluate a label propagation technique to mitigate annotation imbalance, boosting model performance. Our findings reveal that the task remains challenging, with current models showing limited performance. In summary, our contributions are: 1) Formalizing the ODAE task, 2) Providing benchmark comparisons of state-of-the-art methods, and 3) Exploring techniques like label propagation to adapt models to this scenario.

2 RELATED WORK

Offline Temporal Action Localization. The Temporal Action Localization (TAL) task involves predicting the onset and offset frames of an action, with the goal of segmenting the occurrence of actions in the video in an offline setting in which the video is completely observed at inference. Several approaches have been proposed to solve this task (Zhang et al., 2022; Wang et al., 2021a; Wang et al., 2021b; Liu et al., 2024). Notably, ActionFormer (Zhang et al., 2022) adopts a Transformer encoder (Vaswani, 2017), while ActionMamba (Chen et al., 2024) extends this framework by incorporating a Mamba encoder (Gu and Dao, 2023), achieving state-of-the-art performance.

The TAL task shares important similarities with our problem definition, as both aim to detect action instances. However, a major limitation of TAL approaches is their reliance on offline processing, which makes them incompatible with the online constraints of our approach.

Online Action Detection. Online Action Detection (OAD) aims to detect the frames associated with an action as early as possible from partial observations, ideally before the action is completed (De Geest et al., 2016). Since the task is performed in an online setting, predictions at time t' must be made using only observations available at time $t' < t''$. Recent advances in OAD have taken advantage of Transformer-based architectures (Vaswani, 2017), which are well suited to handling long sequences of data. In particular, OadTR (Wang et al., 2021c) is an encoder-decoder framework built on top of Transformers that simultaneously encodes historical information and predicts future context to detect ongoing actions. TeSTra (Zhao and Krähenbühl, 2022), another state-of-the-art model based on transformers, incorporates both long- and short-term memory to effectively summarize past information for improved prediction.

In addition to transformer-based approaches, some work has explored alternative architectures that also yield competitive performance. For example, MiniRoad (An et al., 2023), a fully RNN-based model, achieves similar performance to Transformer-based methods with a smaller memory footprint and increased inference speed. TeSTra-Mamba (Chen et al., 2024) extends TeSTra by replacing the Transformer encoder with a Mamba-based architecture.

While these models are promising and adaptable to the OAD task, they require further refinement to reliably predict an accurate offset frame, as we show in this paper. In addition, proper evaluation using appropriate metrics is essential to assess their performance in this context.

Online Detection of Action Start. The Online Detection of Action Start (ODAS) task (Shou et al., 2018) focuses on accurately predicting the frame at which an action begins in online settings, with an emphasis on temporal accuracy. Previous research has addressed this challenge using various approaches, such as 3D convolutional networks from a third-person perspective (Shou et al., 2018), the combination of LSTMs with reinforcement learning (Gao et al., 2019), and weakly supervised learning techniques using video-level labels (Gao et al., 2021). In addition, recent work has relaxed the online constraint by employing a buffer window to predict the action start frame in a quasi-online fashion (Scavo et al., 2023).

While this is very similar to the problem we aim to address in terms of formulation and evaluation metrics, we argue that for most practical applications predicting the action offset frame is a more practical and sufficient solution.

Datasets of Egocentric Videos. Datasets of egocentric videos (Damen et al., 2022; Grauman et al., 2022; Li et al., 2018; Sener et al., 2022) have received considerable attention in recent years, particularly in the fields of computer vision and human-object interaction. These datasets are characterized by data captured from a first-person perspective, providing valuable insights into how individuals interact with their environment and others. The unique perspective of egocentric data allows for a more refined understanding of context and action, making it particularly useful for studying behavioral patterns and contextual dynamics. A notable example is the use of wearable cameras to record daily activities, providing detailed information about both individual behavior and environmental context.

In this work, we focus on the Epic-Kitchens 100 (EK100) dataset (Damen et al., 2022), a large-scale collection of egocentric video footage that captures a variety of routine kitchen activities. EK100 is a well-established resource for human-object interaction research because it contains detailed video and audio recordings of interactions with kitchen utensils and appliances. Importantly, the dataset also includes complex multitasking scenarios, such as washing dishes while cooking, that involve parallel goal-directed actions. These multitasking interactions present a higher level of difficulty and enrich the applicability of the dataset for studies of human behavior in dynamic, real-world environments.

For the purposes of our study, we adapted the EK100 dataset to our task focusing on video segments representing ”take” and ”release” actions. This allows us to define a new benchmark for take/release temporal action detection.

3 PROBLEM DEFINITION AND EVALUATION METRIC

3.1 Online Detection of Action End (ODAE) Task Definition

We define the ODAE task as follows: given an input video $\mathcal{V}$ observed up to current time t', models have to determine whether the current frame contains the end of a take/release action. Predictions are made online with no access to any frame $t'' > t'$ when making predictions at time t'. Let $a = (c,t)$ represents a ground truth action, where c is the action class and t is the related action end time-stamp. Each prediction made by the model in an online setting is represented as a $\hat{a} = (\hat{c},\hat{t},s)$ tuple, where $\hat{c}$ and $\hat{t}$ are respectively the predicted class and key timestamp and s is a confidence score. Ideally, we aim to obtain a set of high-confidence predictions such as each $\hat{a} = (\hat{c},\hat{t},s)$ prediction is associated to only one ground truth action $a = (c,t)$.

3.2 Evaluation Protocol

In the context of Temporal Action Localization (TAL) and Online Action Detection (OAD), models are typically evaluated using segment-level mean Average Precision (mAP) (Zhang et al., 2022) and frame-level mAP metrics (De Geest et al., 2016; Zhao and Krähenbühl, 2022). Frame-level mAP primarily measures the accuracy of classifying individual frames, while segment-level mAP focuses on the accuracy of detecting action segment boundaries. Neither of these metrics, however, are directly suitable to assess performance in detecting the precise locations of action starts or ends.

Point Level mAP. To properly evaluate our models, as outlined in (Shou et al., 2018), we use point-level detection mean Average Precision (p-mAP), according to which a predicted action $\hat{a} = (\hat{c},\hat{t},s)$ is matched to a ground truth action $a = (c,t)$ if it meets the following criteria:

1. The predicted and ground truth action classes match ($c = \hat{c}$);
2. The temporal offset $\delta = |\hat{t} - t|$ is less than or equal to a specified evaluation temporal threshold ϕ.

Predictions are matched to ground truth actions in a greedy manner, prioritizing those with higher confidence scores. Each predicted or ground truth action can be matched to another action only once. Matched predictions are counted as true positives, unmatched ground truth actions are counted as false positives, whereas unmatched predicted actions are counted as false positives. The mAP value is hence computed averaging AP values for each class following (Caba Heilbron et al., 2015). The final mp-mAP value is hence defined as the average of p-mAP values calculated at different temporal offset thresholds ϕ. In particular we evaluate in a temporal threshold range of 1 to 10 seconds with a step of 1 second.

4 COMPARED METHODS

We benchmark the performance of different methods operating in both offline and online settings. The following sections describe the main considered approaches, for which we test and compare different settings in our experimental analysis.

4.1 ActionFormer

ActionFormer is an encoder-decoder architecture designed for offline video action detection, leveraging a Transformer encoder to encode feature sequences and a 1D convolutional decoder with classification and regression heads to predict action classes and temporal boundaries. As a baseline, ActionFormer was trained on the proposed Offline Detection of Action Ends (ODAE) and Online Detection of Action Start (ODAS) tasks to provide an upper-bound performance reference for online methods.

4.2 TeSTra

TeSTra (Temporal Smoothing Transformer) is a transformer-based model optimized for real-time online action detection and anticipation, incorporating a novel attention mechanism and temporal smoothing kernels to capture long- and short-term dynamics efficiently. By leveraging the box kernel, which operates like a FIFO queue with $O(T)$ space complexity, TeSTra achieves up to 6x faster processing speeds compared to sliding window transformers, enabling real-time action prediction without future frame reliance.

4.2.1 Long-Short Memory

In TeSTra, temporal information from video clips is captured by two different types of memory: long and short memory. The model uses these memory types to enhance its ability to process sequential data. Specifically, the first encoder is used to generate a long memory embedding, which is then passed to the decoder along with the more recent frames, allowing the it to generate the short memory independently. In settings where model efficiency is critical, the role of long memory deserves careful consideration, as it has a direct impact on model size.

To evaluate the utility of long memory in TeSTra in our ODAE setting, we conducted experiments on two datasets with different characteristics: THUMOS14 and EPIC-KITCHENS-100. These datasets differ not only in their nature - THUMOS14 consists of third-person video perspectives and EPIC-KITCHENS-100 contains egocentric footage - but also in the duration of the actions depicted. In the case of THUMOS14, where actions tend to be longer, long memory may be useful for prediction because it provides context over longer periods of time. However, in EPIC-KITCHENS-100, where actions tend to be shorter and more frequent, long memory appears to be less useful. We ablate these aspects in our experiments.

4.2.2 Label Propagation

In the Online Detection of Action End (ODAE) task, data imbalance is a significant challenge, mainly due to the sparse distribution of action ends in untrimmed video. In our context, indeed, models are trained considering the frame marking the end of an action as a positive sample, while all other frames are labeled as negative samples. This approach severely affects the imbalance problem making training dominated by background samples.

To address this issue, we propose an optimized training strategy, inspired by the approach proposed in (Hu et al., 2024), that recognizes the high similarity between frames that occur just before the end of the action. Specifically, we extend the labeling process by including the Δ frames preceding the true action end frame as positive samples. This modification aims to enrich the training data, mitigate the imbalance, and improve the models ability to localize action boundaries.

During training, we propagate the positive label to a maximum of three frames prior to the ground truth action end, in addition to the annotated boundary frame. This strategy provides a compromise that reduces the imbalance in the annotation while minimizing the risk of overfitting to overly long segments, losing the ability to precisely localize action end frames. Although this approach introduces up to three false positives during training (i.e., the model may predict action endings 1-3 frames before the ground truth), in our experiments it does not negatively affect performance during evaluation, as the model is ultimately evaluated only on the accuracy of its predicted action end frame.

4.3 TeSTra - Mamba

Over the past years, Mamba (Gu and Dao, 2023) has emerged as a promising alternative to the Transformer architecture, offering comparable or superior performance while achieving sub-quadratic complexity in both space and time. In particular, Mamba exploits parallelization during training and acts like a recurrent neural network (RNN) during inference, hence offering important computational advantages in online scenarios.

Based on the TeSTra-Mamba model presented in (Chen et al., 2024), we modified the TeSTra architecture replacing the Transformer decoder responsible for short-term memory with a standard Mamba block. Our experiments focused solely on the short-term memory module, excluding any long-term memory components, also considering that Mamba should be able to effectively model observations in a long-term fashion by design.

5 EXPERIMENTAL SETTINGS AND RESULTS

5.1 Datasets

We perform experiments on two benchmark datasets: THUMOS14 (Idrees et al., 2017) and EPIC-KITCHENS-100 (Damen et al., 2022) following the same data settings used in TeSTra (Zhao and Krähenbühl, 2022).

EPIC-KITCHENS-100. The EPIC-KITCHENS-100 dataset contains 100 hours of egocentric video footage with approximately 90,000 annotated action segments. The action labels in EK100 are categorized into 97 verb classes and 300 noun classes. In the EK100 experiments, we ran two main types of experimental conditions. The first condition, termed *all verbs*, required the model to predict all verbs present in the dataset, in addition to the corresponding noun. The second condition, termed *take/release only*, involved reducing the set of target verbs to two categories: “take” and “release”. In this simplified condition, “take” served as a representative for a number of verbs such as “get”, “fetch”, and “collect-from”, while “release” represented verbs such as “put”, “leave-on”, and “place-on”. We used the original EPIC-KITCHENS-100 annotation where we treated the “take” class as it is and the “put” class as our “release”. We evaluated the models ability to predict both verbs and nouns when working in the *all verbs* setting, and only verbs in the *take/release only*.

THUMOS14. The THUMOS14 dataset consists of 413 unedited videos annotated with 20 action categories. We train our model on the validation set, which contains 200 videos, and report performance on the test set, which contains 213 videos. Although the THUMOS14 dataset is exocentric in nature, we used it as a baseline to gain insight into the performance of the tested models and the ability of our task formulation to generalize to the case of third-person observations.

5.2 Results

5.2.1 Offline Detection with ActionFormer

Table 1 reports the results of the offline ActionFormer model trained and tested in ODAS and ODAE settings. Results are reported in percentage. We consider both the case of only Take/Release verbs and all verbs. We note that the ODAE setting brings slightly better results with an mp-mAP of 66.12 as compared to the 65.20 of ODAS settings for Take/Release actions. This suggests that detecting action ends is a less ambiguous task than detecting action starts even in offline settings. Performance values are smaller when all verbs are considered, but they follow a similar trend, with ODAE performing better than ODAS.

Table 1: Offline ActionFormer results in terms of percentage of mp-mAP on EK100 for verb prediction, both with Take/Release only (T/R) and all verbs (All). Best results are reported in bold.

Task	Verbs	Verb mp-mAP
ODAS	T/R	65.20
ODAE	T/R	**66.12**
ODAS	All	25.50
ODAE	All	**25.70**

Table 2: Offline ActionFormer results on THUMOS14 in terms of percentage of mp-mAP.

Task	mp-mAP
ODAS	81.90
ODAE	**83.56**

Results in Table 2 show a similar trend on THUMOS14, with the ODAE setting bringing better results than ODAS. Performance measures achieve larger numbers here due to the simpler nature of the dataset. This highlights that the proposed ODAE formulation is beneficial also in third-person vision settings, reducing ambiguities in action prediction.

5.2.2 TeSTra

EPIC-KITCHENS-100. Table 3 shows the results of different model configurations evaluated in ODAE settings on the EPIC-KITCHENS-100 dataset. We observe that the L/S (Long and Short term) setting without label propagation achieves the best overall performance among models without Mamba layers in the “all verbs” settings, with an average mp-mAP of 6.45 and an action mp-mAP of 5.64. This suggests that combining long-term and short-term memory without label propagation provides a strong baseline for the “all verbs” task. However, when focusing on specific prediction tasks (verb or noun), alternative configurations show superior performance on individual metrics. For example, using 4-frame label propagation and short-term memory (S) configuration achieves the highest verb mp-mAP (6.16) and competitive noun mp-mAP (7.94). Similarly, the using 4-frame label propagation and long-short memory (L/S) achieves the best noun mp-mAP (8.14), but with inferior verb performance (4.90). This suggests that label propagation has different effects depending on the use

Table 3: Performance of TeSTra and TeSTra - Mamba on EPIC-KITCHENS-100 in various scenarios. In the settings column, we indicate whether we used Long memory (L), short memory (S) and the number of Mamba Layers (M) in the TeSTra - Mamba models. Performance are shown as percentage of mp-mAP. Best results are reported in bold.

Action Classes	Settings	Label propagation ($\Delta+1$)	Verb mp-mAP	Noun mp-mAP	Action mp-mAP	Average
All Verbs	L/S	NO	5.95	7.76	**5.64**	**6.45**
All Verbs	L/S	4 Frames	4.90	**8.14**	5.11	6.05
All Verbs	S	NO	6.09	7.20	4.88	6.04
All Verbs	S	4 Frames	**6.16**	7.94	4.62	6.24
All Verbs	S	2 Frames	5.66	7.37	4.70	5.91
All Verbs	S; M:1	NO	7.01	8.05	5.23	6.76
All Verbs	S; M:1	4 Frames	7.98	8.63	5.20	7.27
All Verbs	S; M:1	2 Frames	7.52	8.64	5.35	7.17
All Verbs	S; M:2	NO	**8.41**	**8.66**	**5.36**	**7.48**
All Verbs	S; M:2	4 Frames	7.05	7.82	4.90	6.59
All Verbs	S; M:2	2 Frames	6.79	7.70	4.93	6.47
TR	L/S	NO	20.10	6.39	5.14	10.54
TR	L/S	4 Frames	11.40	5.26	3.06	6.57
TR	S	NO	**20.32**	8.33	**6.09**	**11.58**
TR	S	4 Frames	18.31	**8.64**	5.82	10.92
TR	S	2 Frames	18.76	7.58	5.70	10.68
TR	S; M:1	NO	24.55	**8.97**	**6.65**	**13.39**
TR	S; M:1	4 Frames	21.10	8.02	5.34	11.48
TR	S; M:1	2 Frames	20.86	8.59	6.00	11.81
TR	S; M:2	NO	**25.16**	8.06	6.62	13.28
TR	S; M:2	4 Frames	24.60	7.91	6.15	12.88
TR	S; M:2	2 Frames	19.81	8.05	5.73	11.19

Table 4: TeSTras performance on THUMOS14 in various scenarios. In the settings column, we indicate the presence or not of long memory (L), short memory (S) and the number of Mamba Layers (M). Performance are shown as percentage of mp-mAP. Best results are reported in bold.

Settings	Label propagation ($\Delta+1$)	mp-mAP
L/S	NO	**53.75**
L/S	4 Frames	37.35
S (1s)	NO	37.93
S (1s)	4 Frames	36.77
S; M:2	NO	40.43
S; M:2	4 Frames	**42.12**

of long and short memory. In particular, label propagation improves verb prediction when only the short memory is used, while noun prediction is improved when both long- and short- memory are considered together with label propagation. In general, not using label propagation, but using long-short term memory gives the most balanced results. Adding Mamba layers improve results with best overall results of 8.41/8.66/5.36/7.48 (Verb, Noun, Action, Average) obtained with two mamba layers and no label propagation. Label propagation seems to marginally improve performance with a single Mamba layer, while no label propagation leads to best results when two Mamba layers are considered. This suggests that the use of Mamba layers can mitigate the problem of label sparsity in learning.

We observe similar trends for the "take-release" setting (second half of the table), with best overall results obtained by the Mamba-TeSTra model with 1 Mamba layer and no label propagation (24.55/8.97/6.65/13.39), while methods with no Mamba layers generally achieve lower results. The reason that the best results were obtained with a single Mamba layer may be due to the simpler nature of the task. Verb accuracy in particular greatly benefits from the Mamba layers. Indeed, the best Mamba-TeSTra architecture obtains a verb accuracy of 25.16 versus the 20.32 of the best TesTRa architecture, suggesting that Mamba enables better temporal modeling of features allowing for stronger motion recognition useful to recognize take and release actions.

THUMOS14. Table 4 shows the performance of the models on the THUMOS14 dataset. The results show that the best performing configuration is the model using long-term and short-term memory and no Mamba layers, which achieves an mp-mAP of 53.75, versus 42.12 in the best Mamba-TeSTra configuration. It is worth noting that, while Mamba layers are not helpful in this dataset, Mamba-TeSTra architectures still outperform models using only the short memory. This suggests that 1) Mamba layers bring added value with respect to only using short memory, and that 2) the dataset does not require complex and scalable past

Table 5: Ablation studies on the contribution of each feature used.

Used Feature	Verb	Noun	Action	Average
RGB	20.08	7.10	5.22	10.08
Optical Flow	**25.00**	4.29	2.80	10.69
RGB + Optical	24.55	**8.97**	**6.65**	**13.39**

encoding mechanisms such as Mamba, probably due to THUMOS14 actions being longer and less densely annotated than the ones in EPIC-KITCHENS. Also in this case, label propagation seems to bring minor benefits only in specific settings (e.g., with Mamba-TeSTra).

5.2.3 Ablation Studies

To deepen our understanding of the task, we conducted ablation studies assessing the specific contributions of each feature type to the overall performance. Our models take as input concatenated RGB and Optical Flow features. The two signals encode specific properties of the input. For instance, RGB images encode appearance, whereas Optical Flow encodes object motion. To assess the contribution of optical flow to final performance, we ran three experiments where RGB, Optical Flow and the concatenation of both where used within the overall best performing model on the EPIC-KITCHENS-100 dataset, which uses two Mamba layers without label propagation. Experiments are in the "take-release" settings here. The results are shown in Table 5. We note that optical flow provides the most critical information for verb prediction, at the cost of losing some information necessary for noun prediction. Best average performance is obtained by using both RGB and optical flow.

Indeed, using optical flow alone achieves the best verb prediction performance, outperforming the combination of RGB and optical flow features. Interestingly, while the RGB-only model shows lower verb performance (4.47 lower mp-mAP) compared to the RGB + Optical Flow setting, it delivers comparable noun prediction accuracy (1.87 lower mp-mAP) and offers significant advantages in inference efficiency by eliminating the computationally expensive optical flow calculations. This trade-off highlights the flexibility of RGB-only models for real-time applications while showcasing the potential of optical flow for tasks prioritizing verb recognition accuracy.

6 CONCLUSIONS

This work introduces the Online Detection of Action End (ODAE) task, which focuses on real-time detection of action endpoints, such as ”take” and ”release,” in egocentric video analysis. Using the EPIC-KITCHENS-100 and THUMOS14 datasets, we benchmarked state-of-the-art temporal action detection models, finding that traditional methods struggle with the stringent temporal accuracy and efficiency requirements of ODAE, particularly in dense action scenarios. Transformer-based models like TeSTra exhibited limitations, while Mamba-based architectures showed significant improvements due to their efficient temporal modeling. Label propagation techniques were explored to address annotation imbalances caused by sparse action endpoints, yielding measurable accuracy improvements, especially in short-memory configurations. Analysis also revealed that short-term memory and combined RGB-optical flow features are crucial for capturing the immediate context of short, rapid actions. This study formalizes ODAE, evaluates existing models, and highlights strategies for improving online action endpoint detection in challenging scenarios.

ACKNOWLEDGEMENTS

This research has been supported by the project EXTRA-EYE - PRIN 2022 - CUP E53D23008280006 - Finanziato dall'Unione Europea - Next Generation EU, Missione 4 Componente 1 CUP E53D23008280006.

REFERENCES

An, J., Kang, H., Han, S. H., Yang, M.-H., and Kim, S. J. (2023). Miniroad: Minimal rnn framework for online action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10341–10350.

Caba Heilbron, F., Escorcia, V., Ghanem, B., and Carlos Niebles, J. (2015). Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970.

Chen, G., Huang, Y., Xu, J., Pei, B., Chen, Z., Li, Z., Wang, J., Li, K., Lu, T., and Wang, L. (2024). Video mamba suite: State space model as a versatile alternative for video understanding. *arXiv preprint arXiv:2403.09626*.

Cheng, T., Shan, D., Hassen, A., Higgins, R., and Fouhey, D. (2023). Towards a richer 2d understanding of hands at scale. *Advances in Neural Information Processing Systems*, 36:30453–30465.

Damen, D., Doughty, H., Farinella, G. M., Furnari, A., Kazakos, E., Ma, J., Moltisanti, D., Munro, J., Perrett, T., Price, W., et al. (2022). Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision*, pages 1–23.

Darkhalil, A., Shan, D., Zhu, B., Ma, J., Kar, A., Higgins, R., Fidler, S., Fouhey, D., and Damen, D. (2022). Epic-kitchens visor benchmark: Video segmentations and object relations. *Advances in Neural Information Processing Systems*, 35:13745–13758.

De Geest, R., Gavves, E., Ghodrati, A., Li, Z., Snoek, C., and Tuytelaars, T. (2016). Online action detection. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 269–284. Springer.

Gao, M., Xu, M., Davis, L. S., Socher, R., and Xiong, C. (2019). Startnet: Online detection of action start in untrimmed videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5542–5551.

Gao, M., Zhou, Y., Xu, R., Socher, R., and Xiong, C. (2021). Woad: Weakly supervised online action detection in untrimmed videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1915–1923.

Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al. (2022). Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012.

Gu, A. and Dao, T. (2023). Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Hu, X., Wang, S., Li, M., Li, Y., and Du, S. (2024). Time-attentive fusion network: An efficient model for online detection of action start. *IET Image Processing*, 18(7):1892–1902.

Idrees, H., Zamir, A. R., Jiang, Y.-G., Gorban, A., Laptev, I., Sukthankar, R., and Shah, M. (2017). The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23.

Li, Y., Liu, M., and Rehg, J. M. (2018). 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.

Liu, S., Zhang, C.-L., Zhao, C., and Ghanem, B. (2024). End-to-end temporal action detection with 1b parameters across 1000 frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18591–18601.

Plizzari, C., Goletto, G., Furnari, A., Bansal, S., Ragusa, F., Farinella, G. M., Damen, D., and Tommasi, T. (2024). An outlook into the future of egocentric vision. *International Journal of Computer Vision*, pages 1–57.

Scavo, R., Ragusa, F., Farinella, G. M., and Furnari, A. (2023). Quasi-online detection of take and release actions from egocentric videos. In *International Conference on Image Analysis and Processing*, pages 13–24. Springer.

Sener, F., Chatterjee, D., Shelepov, D., He, K., Singhania, D., Wang, R., and Yao, A. (2022). Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21096–21106.

Shan, D., Geng, J., Shu, M., and Fouhey, D. F. (2020). Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9869–9878.

Shou, Z., Pan, J., Chan, J., Miyazawa, K., Mansour, H., Vetro, A., Giro-i Nieto, X., and Chang, S.-F. (2018). Online detection of action start in untrimmed, streaming videos. In *Proceedings of the European conference on computer vision (ECCV)*, pages 534–551.

Souček, T., Alayrac, J.-B., Miech, A., Laptev, I., and Sivic, J. (2022). Look for the change: Learning object states and state-modifying actions from untrimmed web videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13956–13966.

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

Wang, X., Qing, Z., Huang, Z., Feng, Y., Zhang, S., Jiang, J., Tang, M., Gao, C., and Sang, N. (2021a). Proposal relation network for temporal action detection. *arXiv preprint arXiv:2106.11812*.

Wang, X., Zhang, S., Qing, Z., Shao, Y., Gao, C., and Sang, N. (2021b). Self-supervised learning for semi-supervised temporal action proposal. In *CVPR*.

Wang, X., Zhang, S., Qing, Z., Shao, Y., Zuo, Z., Gao, C., and Sang, N. (2021c). Oadtr: Online action detection with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7565–7575.

Xue, Z., Ashutosh, K., and Grauman, K. (2024). Learning object state changes in videos: An open-world perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18493–18503.

Zhang, C.-L., Wu, J., and Li, Y. (2022). Actionformer: Localizing moments of actions with transformers. In *European Conference on Computer Vision*, pages 492–510. Springer.

Zhao, Y. and Krähenbühl, P. (2022). Real-time online video detection with temporal smoothing transformers. In *European Conference on Computer Vision*, pages 485–502. Springer.