SlowFast Rolling-Unrolling LSTMs for Action Anticipation in Egocentric Videos

Nada Osman, Guglielmo Camporese, Pasquale Coscia, Lamberto Ballan
Department of Mathematics "Tullio Levi-Civita"
University of Padova, Italy

{nadasalahmahmoud.osman, guglielmo.camporese}@phd.unipd.it

{pasquale.coscia, lamberto.ballan}@unipd.it

Abstract

Action anticipation in egocentric videos is a difficult task due to the inherently multi-modal nature of human actions. Additionally, some actions happen faster or slower than others depending on the actor or surrounding context which could vary each time and lead to different predictions. Based on this idea, we build upon RULSTM architecture, which is specifically designed for anticipating human actions, and propose a novel attention-based technique to evaluate, simultaneously, slow and fast features extracted from three different modalities, namely RGB, optical flow and extracted objects. Two branches process information at different time scales, i.e., frame-rates, and several fusion schemes are considered to improve prediction accuracy. We perform extensive experiments on EpicKitchens-55 and EGTEA Gaze+ datasets, and demonstrate that our technique systematically improves the results of RULSTM architecture for Top-5 accuracy metric at different anticipation times.

1. Introduction

Human action anticipation [4, 11, 34] is a popular research topic in computer vision due to a wide range of involved applications. For example, assistive robotic platforms [17, 28] need to anticipate human movements to correctly perform their tasks when multiple people are present in the same environment. Similarly, advanced video-surveillance systems [20] require to anticipate human motion to promptly provide timely assistance. In this context, egocentric videos have provided a considerable amount of information to be used for training action anticipation models thanks to low-cost wearable devices which offer different streams to be used [22, 31], *e.g.*, RGB videos, audio or depth data.

State-of-the-art approaches [29, 35] are mainly based on attention mechanisms to efficiently extract relationships across subsequent frames at a specific frame rate. Neverthe-
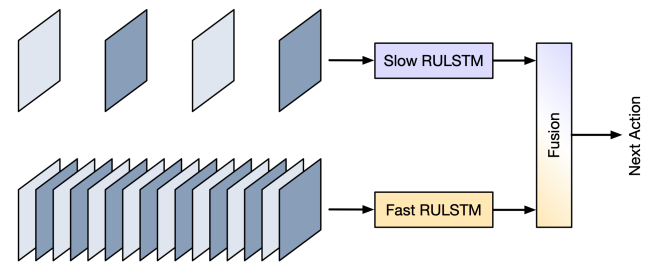

Figure 1: Human actions happen at different speeds requiring a multi-scale approach for better predicting future behaviors. We propose a Slow-Fast RU-LSTM model containing two branches, namely *slow* and *fast* branch, which learn independently from input videos features at different time scales.

less, action speed may differ based on the actor, surrounding environment and action itself.

To anticipate future actions, two main factors should be taken into account: window size (*i.e.*, number of current and past actions to be considered) and processing frame rate (*i.e.*, quantity of information to be extracted from each action). While the former is typically fixed for a fair results comparison, the latter can be arbitrary selected. In this case, a different choice of this parameter may lead to completely different results. We demonstrate that, if multiple streams of the same modality is provided to an action anticipation model, it is able to appropriately select which stream to focus on and improve its predictive capabilities leading to a better generalization.

Based on this idea, we propose to consider multiple branches for each input modality which process the corresponding stream at different frame rates. We focus on two popular egocentric datasets, namely EPIC-Kitchens-55 and EGTEA GAZE+. Based on RU-LSTM [13] model, we propose a slow-fast architecture that learns from input

videos at two different scales, as shown in Figure 1. A slow branch processes input videos with a small frame rate while another branch uses a higher frame rate. In this way, redundant information is discarded for actions that evolve slowly while retained for faster actions. In order to efficiently combine these two branches, we use an attentive-based mechanism which efficiently weights their output scores and provides only one result which is subsequently decoded to extract future actions. We show that our model systematically outperforms state-of-the-art models at different anticipation times.

The main contributions of our work can be summarized as follows:

1. We propose a multi-scale learning technique that benefits from a slow and fast branch to augment performance of RU-LSTM model;

2. We perform extensive ablation experiments in order to select the most appropriate frame rates and window sizes;

3. We conduct multiple evaluation experiments on popular action anticipation benchmarks and also compare different model architectures and slow-fast fusion mechanisms.

2. Related Work

2.1. Action Recognition

Action recognition consists of predicting a labelled action category assigned to an input video. Learning from videos requires capturing both spatial and temporal information, and several approaches have been proposed to solve this task. A simple modelling strategy is based on extracting spatial features from observed video frames with a 2D Convolutional Neural Network (CNN) and their aggregation at temporal level [16, 24], or with Long-Short Term Memory (LSTM) networks [6, 24]. Another popular approach exploits 3D CNNs where spatio-temporal information is gradually fused, leading to a better video representation and more accurate results [3, 33, 15, 32]. Another successful idea uses two-stream networks where RGB frames and optical flow features are processed providing a more detailed motion information contained in input videos [30, 9, 3, 24].

Recognizing an observed action is the first step for solving more complex tasks, such as early action recognition, where a future action is predicted using only a partial observation of the input video, and action anticipation, where an action category is predicted using only past observed frames.

2.2. Action Anticipation

Action anticipation requires to predict future actions relying only on past video frames [14]. Previous works proposed different models for activity anticipation in third-person videos [1, 10, 14, 18, 21, 36] and first-person videos [7, 12, 27, 37, 2]. In our work, we adopt the formulation presented in [13], where an action to be predicted is computed at fixed anticipation times before it starts. This is a challenging task since it involves learning both spatial and temporal relationships among past and future frames. To this end, [13] proposes an encoder-decoder LSTM-based architecture where past information is firstly summarized, and future actions are then computed leveraging features extracted from past information.

2.3. Multi-Scale Modelling in Vision

Multi-scale modelling is a powerful design paradigm that empowers a hidden input representation to be more robust to scale changes with respect to a single-scale modelling approach. This technique can be adopted in both spatial [23, 25] and temporal [8, 29] domains. Slow-Fast networks [8] for video recognition builds upon this idea and show to benefit from processing video sequences at slow and fast frame rates with two separate branches that capture patterns at different time resolutions. In our work, we take advantage of this idea aiming at capturing slow and fast features for anticipating future actions.

3. Proposed Method

Action anticipation consists of predicting future actions using only visual information extracted from current and past frames. As proposed in [13], a future action is predicted at the anticipation time of $1 \ s$ before it occurs, and only video information before this anticipation time can be used for its prediction. More specifically, the evaluation protocol requires to anticipate the future action at subsequent time steps in order to evaluate the performance of the model approaching the target action.

In the following, we briefly summarize our baseline, which constitutes the backbone used at different time granularities, and then present our slow-fast fusion technique. Finally, we describe how our slow-fast approach can be both used for one input modality and multiple modalities using modality attention [13].

3.1. Rolling-Unrolling LSTM

Our technique is built upon RU-LSTM [13] model, which processes sequences of feature vectors computed from input video frames. This model defines an encoding stage of S_{enc} steps and an anticipation stage of S_{ant} steps for a total of $\alpha \cdot (S_{enc} + S_{ant})$ seconds, where α is the time interval between two subsequent frames. This model

is based on an LSTM-based encoder, named *rolling* LSTM (R-LSTM), and an LSTM-based decoder, named *unrolling* LSTM (U-LSTM). The former summarizes, during the encoding and anticipation stages, past information extracted from input videos and provides to the latter a useful context for predicting the future action. The decoder, in the anticipation stage, receives the representation from the encoder and, using the last observation, computes a plausible distribution over future action classes. The encoding-decoding process is performed for each time step in the anticipation stage, and the network is trained for predicting the actual action label using a cross-entropy loss. To exploit more context and create a more informative hidden representation, RU-LSTM processes multi-modal features which are combined using a mixture-of-experts-based method named Modality Attention (MATT). Since this model shows remarkable performance on predicting future actions from multi-modal input streams, we extend its predictive capability by explicitly designing a multi-scale fusion mechanism able to capture slow and fast features from observed video sequences.

3.2. SlowFast RULSTM

As depicted in Figure 1, our SlowFast RULSTM model consists of two branches: a slow branch, that processes input videos using a low frame rate (one frame every α_s seconds), and a fast branch, which uses a high frame rate (one frame every α_f seconds). Our idea is to process input features at different time resolutions in order to capture slow and fast relations between past and future frames.

Let $\boldsymbol{x} \in \mathbb{R}^{T \times C \times H \times W}$ be the input video to be processed and $\boldsymbol{z} \in \mathbb{R}^{T \times D}$ the corresponding representation computed at each time step. Given a single input frame $\boldsymbol{x}_t$ at time t, $\boldsymbol{z}_t = \phi(\boldsymbol{x}_t)$ is its related representation where ϕ is a CNN feature extractor, and $T = S_{enc} + S_{ant}$ the total sequence length. Our slow branch processes input video frames at $1/\alpha_s$ frame rate while our fast branch at $1/\alpha_f = R/\alpha_s$ with $R = \alpha_s/\alpha_f$ being the ratio between fast and slow frame rates, respectively. Given an internal representation $\boldsymbol{z}_t$, the encoder in the fast branch produces feature representations used by the decoder as follows:

$$\boldsymbol{r}_t^f = FR\text{-}LSTM \left(\boldsymbol{z}_t, \boldsymbol{r}_{t-1}^f \right) \qquad (1)$$

where $t \in \{1, 2, \ldots, T\}$ and $\boldsymbol{r}_t^f = (\boldsymbol{h}_t^f, \boldsymbol{c}_t^f)$ is the state that contains hidden and context vectors of FR-LSTM with $\boldsymbol{h}_t^f, \boldsymbol{c}_t^f \in \mathbb{R}^d$. Our slow branch is similarly defined:

$$\boldsymbol{r}_t^s = SR\text{-}LSTM \left(\boldsymbol{z}_t, \boldsymbol{r}_{t-1}^s \right) \qquad (2)$$

where $t = kR + 1$ with $k \in \{0, 1, \ldots, \lfloor T/R \rfloor\}$, and $\boldsymbol{r}_t^s = (\boldsymbol{h}_t^s, \boldsymbol{c}_t^s)$ is the state containing hidden and context vectors of slow R-LSTM with $\boldsymbol{h}_t^s, \boldsymbol{c}_t^s \in \mathbb{R}^d$. The decoder in the fast branch receives the representations given by the fast

encoder and produces the prediction by unrolling the Fast U-LSTM for $T - t + 1$ steps as follows:

$$\boldsymbol{u}_{t,q}^f = FU\text{-}LSTM \left(\boldsymbol{z}_t, \boldsymbol{u}_{t,q-1}^f \right), \qquad (3)$$

$$\boldsymbol{u}_{t,t-1}^f = \boldsymbol{r}_t^f, \qquad \boldsymbol{u}_t^f = \boldsymbol{u}_{t,T}^f, \qquad (4)$$

where $q \in \{t, \ldots, T\}$. Then, a fast prediction score over all action classes is computed from the output of the decoder with a Multi-Layer Perceptron (MLP) at each time step as $\boldsymbol{l}_t^f = MLP(\boldsymbol{u}_t^f)$, where hidden and context vectors in $\boldsymbol{u}_t^f$ are concatenated. Similarly, the slow decoder receives the slow encoded features $\boldsymbol{r}_t^s$ and produces $\boldsymbol{u}_t^s$ by unrolling the Slow U-LSTM and then slow logits $\boldsymbol{l}_t^s$ are computed with a MLP. The formulation related to the slow decoding step is as follows:

$$\boldsymbol{u}_{t,q}^s = SU\text{-}LSTM \left(\boldsymbol{z}_t, \boldsymbol{u}_{t,q-1}^s \right), \qquad (5)$$

$$\boldsymbol{u}_{t,t-1}^s = \boldsymbol{r}_t^s, \qquad \boldsymbol{u}_t^s = \boldsymbol{u}_{t,T}^s, \qquad (6)$$

$$\boldsymbol{l}_t^s = MLP \left(\boldsymbol{u}_t^s \right). \qquad (7)$$

After slow and fast logits scores computation, our model fuses the obtained predictions with an attention mechanism. Specifically, given both slow and fast scores ($\boldsymbol{l}_t^s$ and $\boldsymbol{l}_t^f$), we compute our final merged logits as $\boldsymbol{l}_t = w_t^s \cdot \boldsymbol{l}_t^s + w_t^f \cdot \boldsymbol{l}_t^f$, where w_s and w_f represents slow and fast multipliers that weight slow and fast predictions computed as follows:

$$\left[\lambda_t^s, \lambda_t^f \right] = \text{MLP} \left(\left[\boldsymbol{r}_t^s, \boldsymbol{r}_t^f \right] \right), \qquad (8)$$

$$w_t^s = \frac{e^{\lambda_t^s}}{e^{\lambda_t^s} + e^{\lambda_t^f}}, \qquad w_t^f = \frac{e^{\lambda_t^f}}{e^{\lambda_t^s} + e^{\lambda_t^f}}, \qquad (9)$$

where $\left[\cdot \right]$ stands for the concatenation operator.

3.3. SlowFast and Modalities Fusion Strategies

As proposed in [13], anticipating future actions can take advantage of multi-modal input representations. For this reason, RU-LSTM proposes an attention mechanism (MATT module) that properly weights each input modality. In our work, we exploit the multi-modal video representation and investigate two different techniques to embed both multi-modal and multi-scale inputs. As shown in Figure 3, we could either merge our modalities with a MATT module and then fuse both slow and fast branches (Fig. 3a), or firstly fuse slow and fast branches for each modality, and then merge with a MATT module the multi-modal representations (Fig. 3b). More specifically, Figure 3a depicts an architecture that fuses two RU-LSTMs trained on two different time scales with our slow-fast attention scheme. The input of the attention network is the concatenation of

Figure 2: Our SlowFast RULSTM model. Input videos are firstly processed by a CNN feature extractor and then sequence representations are fed to two branches processing information at two different frame rates. Our slow and fast branches are based on RU-LSTM architecture that encodes past information and then decodes future actions. To better capture the correlations in past observed frames, we design a slow-fast fusion mechanism that merges the predictions of these two branches leading to a better accuracy.

(a) Mod-SF Fusion   (b) SF-Mod Fusion

Figure 3: SlowFast and Modalities fusion schemes. (a) Modalities fusion is applied at slow and fast frame rates and then SlowFast fusion is applied to the fused modalities. (b) SlowFast fusion is firstly applied to each modality separately, and then the modalities fusion is applied to the fused time scales.

the time scale branches, where each branch is represented by the weighted internal representation r_t of the R-LSTM encoders for all the modalities, using the pre-trained modalities attention weights.

As discussed in Sec 3.2, in Figure 3b each modality is trained with a slow and fast branch, fused with the slow and fast module and then each modality is merged with the same MATT used in RU-LSTM.

4. Experimental Results

We conduct several experiments on two popular datasets used for action anticipation in order to investigate our Slow-Fast RULSTM model. Furthermore, we study two architec-

tures that embed different fusion mechanisms dealing with multi-modal and multi-scale inputs. In the following, we describe our datasets, metrics and performed experiments in order to show the impact of our slow and fast modelling approach.

Datasets We experiment on two popular egocentric datasets: EpicKitchens-55 [5] and EGTEA Gaze+ [19]. EpicKitchens-55 collects 55 hours of recorded videos and 39,596 annotations of 32 participants involved in their daily kitchen activities. The annotations contain 125 verb and 352 noun classes. All unique (*verb*, *noun*) pairs are considered for a total of 2,513 unique action labels. EGTEA Gaze+ contains 28 hours of video clips showing hand-object interaction actions performed by 32 participants. It contains 19 verbs, 51 nouns and 106 unique actions. The average across three splits reported by the authors of the dataset is considered.

Evaluation Metric For both datasets, we evaluate our proposed SlowFast RULSTM model using Top-5 accuracy metric at different anticipation times.

Implementation Details We use PyTorch [26] for our implementation and use pre-extracted features provided by [13] for training our method. We found beneficial to train each branch separately and then fine-tuning at the fusion stages. Specifically, for Mod-SF Fusion approach (see 3a), we train RU-LSTM at different frame rates using its standard training pipeline and then fine-tune slow and fast branches at the final stage. For SF-Mod Fusion approach, we apply a similar training strategy.

Top-5 ACTION Accuracy% @ different τ_a(s)				
		2.0	1.5	1.0	0.5
RGB	RULSTM[13]	25.44	28.32	30.83	33.31
	SF-RULSTM	**26.78**	**29.25**	**32.05**	**34.34**
	Imp.	+1.34	+0.93	+1.22	+1.03
FLOW	RULSTM[13]	17.38	18.91	21.42	23.49
	SF-RULSTM	**18.01**	**19.82**	**22.36**	**24.15**
	Imp.	+0.63	+0.91	+0.94	+0.66
OBJ	RULSTM[13]	24.56	26.61	29.89	31.82
	SF-RULSTM	**25.61**	**27.64**	**30.8**	**32.15**
	Imp.	+1.05	+1.03	+0.91	+0.33
FUSION	RULSTM[13]	29.44	32.24	35.32	37.37
	SF-RULSTM	**30.58**	**32.83**	**36.09**	**37.87**
	Imp.	+1.14	+0.59	+0.77	+0.5

Table 1: Top-5 accuracy at different anticipation times for RU-LSTM and our SF-RULSTM model.

4.1. Quantitative Results

Evaluation Results on EpicKitchens-55   Table 1 reports our results for SlowFast RULSTM and RU-LSTM models on EpicKitchens-55 dataset. Our method outperforms RU-LSTM considering both each modality separately and their fusion. The RGB branch shows an improvement of 1.22% at 1 s. Additionally, almost 1% of improvement is achieved for both FLOW and OBJ modalities. Our model, combining all modalities, achieves a 36.09% anticipation accuracy at 1 s, with an improvement of approximately 0.8% over RU-LSTM baseline. Our model also shows a remarkable gain at 2 s of 1.14% validating our idea to use a multi-scale approach for capturing more information at the early stages of action anticipation. Our results prove that processing egocentric videos at different frame rates improves the prediction accuracy.

Table 2 reports a comparison between SlowFast RUL-STM and Temporal Aggregation Block (TAB) models, as proposed in [29], which is a current state-of-the-art multi-scale approach for action anticipation. We report results at anticipation accuracy of 1 s, as authors do not provide anticipation accuracy at different anticipation times. TAB performance is obtained by using the same configuration reported in [29]. Our results show an accuracy improvement for both RGB and FLOW modalities of +3.8% and +2.76%, respectively. In this case, our improvement for both OBJ modality and complete model is less marked, yet our slow-fast fusion model still outperforms TAB model.

Evaluation Results on EGTEA Gaze+   Table 3 compares our proposed SlowFast RULSTM model to RU-LSTM model on EGTEA Gaze+ dataset. For this dataset, only RGB and optical flow features are available, and we train RU-LSTM model to obtain results for both modalities. By contrast, RU-LSTM fusion results are reported from [13]. The table shows a maximum improvement for

Top-5 ACTION Accuracy% @ 1s				
	RGB	FLOW	OBJ	FUSION
TAB	28.25	19.60	30.09	35.73
SF-RULSTM	**32.05**	**22.36**	**30.8**	**36.09**
Imp.	+3.8	+2.76	+0.71	+0.36

Table 2: Comparison of action anticipation Top-5 accuracy at 1 s between SF-RULSTM and TAB [29] model.

Top-5 ACTION Accuracy% @ different τ_a(s)				
		2.0	1.5	1.0	0.5
RGB	RULSTM	56.41	60.68	66.76	72.04
	SF-RULSTM	**57.84**	**62.36**	**67.21**	**72.32**
	Imp.	+1.43	+1.68	+0.45	+0.28
FLOW	RULSTM	33.92	35.83	39.51	42.62
	SF-RULSTM	**36.93**	**39.29**	**42.84**	**45.94**
	Imp.	+3.01	+3.46	+3.33	+3.32
FUSION	RULSTM[13]	56.82	**61.42**	66.4	71.84
	SF-RULSTM	**57.48**	61.37	**67.6**	**72.22**
	Imp.	+0.66	-0.05	+1.2	+0.38

Table 3: Top-5 accuracy at different anticipation times for EGTEA Gaze+ dataset.

the FLOW modality of approximately +3.5% at 1 s. Furthermore, our complete model improves the anticipation accuracy at 1 s by 1.2%, which can be considered a relevant gain due to the reduced number of classes of this dataset compared to EpicKitchens-55 (106 instead of 2513 classes).

4.2. Ablation Experiments on EpicKitchens

To assess the performance of each part of our model, we conduct a set of ablative experiments. In this case, we focus on EpicKitchens-55 dataset. Additionally, all single modality-related experiments use only RGB features, as they can be assumed to be more inclusive features than both optical flow and object-based features.

Selection of Time Step Value   The main element of our model is represented by the choice of slow and fast time steps to be fused.   Table 4 illustrates our anticipation accuracy using different time steps ($\alpha \in \{0.1, 0.2, 0.25, 0.5, 1.0\}$) for RGB features. As shown, the best results (at 1 s) are obtained selecting $\alpha = 0.125$ s and $\alpha = 0.5$ s. For this reason, we use these two values for our fast and slow branches, respectively.

Additionally, Figure 4 compares Top-5 accuracy results, for each modality, using three different time steps: 0.125 and 0.5, as obtained by our previous experiments for the RGB modality, and 0.25, which represents the default time step value used in [33]. As shown, our selected time steps improve Top-5 accuracy for each modality.

Sequence Length Encoding   Extracting relevant features from a video sequence may not only depend on the se-

	Top-5 ACTION Accuracy% @ different τ_a(s)							
α (s)	2.0	1.75	1.5	1.25	1.0	0.75	0.5	0.25
0.1	25.13	-	27.26	-	30.44	-	33.27	-
0.125	24.53	25.63	27.3	**28.97**	**30.96**	32.23	33.49	35.02
0.2	25.16	-	-	-	30.71	-	-	-
0.25	25.2	**25.84**	27.78	28.84	30.55	31.92	33.19	34.43
0.5	**26.39**	-	**28.4**	-	**30.94**	-	32.87	-
1.0	25.56	-	-	-	30.13	-	-	-

Table 4: Top-5 accuracy at different time steps (α) for a single modality (RGB). At 1 s the best performance is achieved considering two frame rates: 0.125 and 0.5.
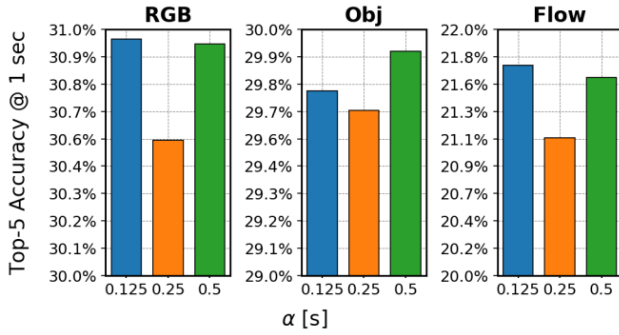
Figure 4: Top-5 accuracy varying the time step α for different input modalities. We select $\alpha \in \{0.125, 0.5\}$ for our SlowFast architecture as each branch appears more accurate with respect to selecting $\alpha = 0.25$, as used in [13].

	Top-5 ACTION Accuracy% @ 1s	
τ_e (s)	$\alpha = 0.125$	$\alpha = 0.5$
1.5	**30.96**	30.94
3.0	30.66	**31.44**

Table 5: Action anticipation results at 1 s for two different lengths of encoding time (τ_e) for RGB modality.

lected frame rate but also on the length of observed sequences. To this end, we test the impact of different buffer lengths on the anticipation task for the RGB features. Two buffer lengths are considered: $\tau_e = 1.5$ s, as proposed in [13], and $\tau_e = 3.0$ s. As shown in Table 5, increasing the buffer length provides a noticeable improvement for the slow model ($\alpha = 0.5$ s), while the opposite arises for the fast model ($\alpha = 0.125$ s). Since the slow model processes a smaller number of video frames, it seems to be able to store more past frames. By contrast, increasing the buffer of the fast model increases its complexity, requiring a smaller window size to achieve better results.

SlowFast Fusion　Table 6 reports our results for different slow-fast fusion schemes considering the RGB modality. The first three rows shows different fusion methodologies

	Top-5 ACTION Accuracy% @ different τ_a(s)				
	α(s)	2.0	1.5	1.0	0.5
Concat	$\{0.125, 0.5\}$	24.59	26.9	30.04	32.73
Ensemble (AVG)	$\{0.125, 0.5\}$	26.98	29.59	31.71	34.2
Attention	$\{0.125, 0.5\}$	26.78	29.25	**32.05**	34.34
Attention	$\{0.125, 0.25, 0.5\}$	26.84	29.51	31.91	33.96

Table 6: Top-5 accuracy at different anticipation times for different slow-fast fusion schemes (RGB modality).

using two scale-branches: slow (with $\alpha = 0.5s$) and fast (with $\alpha = 0.125s$). We consider two additional fusion techniques other than the proposed attention-based fusion:

- *Concat*: prediction obtained directly from the concatenation of the internal representations of the slow and fast branches;

- *Ensemble*: average of the predictions of the slow and fast branches.

As shown, the best fusion scheme at 1 s is represented by an attention-based approach, which appears to better discriminate which branch should be used more for predicting future actions. The last row reports our results for the attention-based model considering an additional scale-branch ($\alpha = 0.25$ s). These results confirm that anticipating future human actions requires different time scales for obtaining better performance. Among the proposed models, best results are achieved using two scale branches (slow and fast), while adding another branch does not provide any improvement.

Modalities Fusion　To assess the performance of the proposed modalities fusion mechanism, shown in Figure 3a (Mod-SF Fusion), an alternative fusion architecture (SF-Mod Fusion) is tested (see Figure 3b). Table 7 provides Top-5 accuracy for both Mod-SF Fusion and SF-Mod Fusion approaches. Additionally, we change the input of the slow-fast attention to be the concatenation of the internal representations of all R-LSTM branches, instead of the weighting mechanism, as discussed in 3.3. As shown, Mod-SF Fusion approach appears the best configuration, since it is easier, compared to the other models, to combine different modalities and allow them to aid each other. Using SF-Mod Fusion, the combination of multi-modal predictions is more complex, and reduces model performance. The approach based on the concatenation, provides the lowest accuracy, which can be due to the huge input size to the attention network.

4.3. Qualitative Results

We qualitatively evaluate the behaviour of our proposed SF-RULSTM in Figure 5 and Figure 6. Figure 5 shows
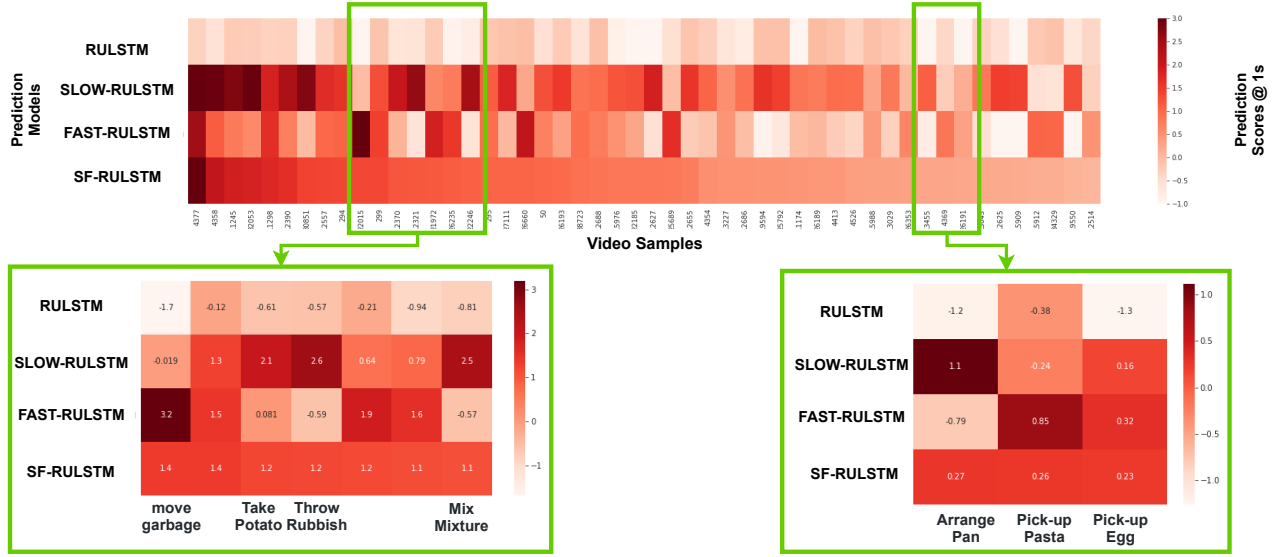
Figure 5: Predictions scores of different video samples from our validation set, where our model provides higher prediction scores than RU-LSTM model. For many actions (*e.g.*, *move garbage, arrange pan*) at least one slow/fast branch has a higher prediction score, and so our complete slow-fast model compared to the selected baseline.
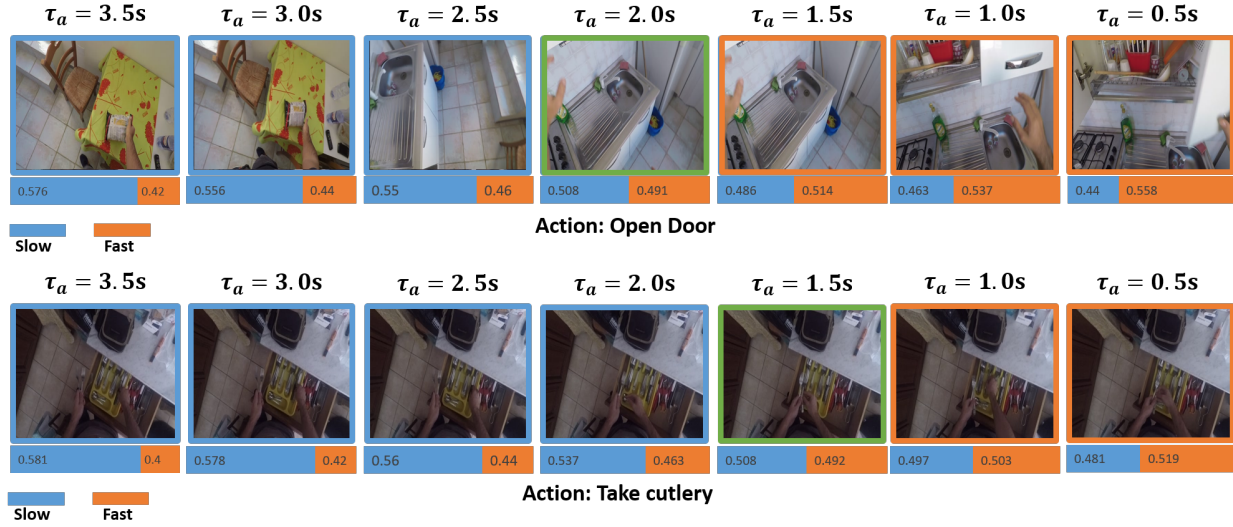

Figure 6: Two examples of actions where slow-fast attention weights change over time. The actions start with no significative changes in the input frames, so the attention mechanism weights more the slow branch. When the action rapidly evolve, more attention is instead provided to the fast branch.

the prediction scores of our SF-RULSTM model (last row) against RU-LSTM model scores (first row) considering a subset of validation samples, *i.e.*, the ones where RU-LSTM assigns low scores. By contrast, our model benefits from either slow (second row) or fast branch (third row) and results in a higher score.

Finally, Figure 6 shows how the slow-fast attention model adapts to different action speeds. Our model is able

to select the most appropriate branch for the current action speed, *i.e.*, the slow one, when limited changes in the RGB video stream occur, or the fast branch for actions that evolve more rapidly.

5. Conclusion

This work proposes a multi-scale attention-based approach to fuse information extracted at different time scales

Top-5 ACTION Accuracy% @ 1s	
Concatenation	31.92
Mod-SF Fusion	**36.09**
SF-Mod Fusion	35.28

Table 7: Top-5 ACTION accuracy at $1 s$ for different variations of modalities fusion.

for anticipating human actions in egocentric videos. Two branches process input videos to capture slow and fast features and better discriminate among different actions (or same action performed by different actors). We design several fusion techniques for combining multiple input modalities and demonstrate that an anticipation model can benefit from fusing input modalities before combining different time scales.

We outperform a state-of-the-art model on two popular benchmarks, *e.g.*, EpicKitchens-55 and EGTEA GAZE+ and show better results compared to a multi-scale model on EpicKitchens-55 dataset. Our future work will focus on considering more branches and investigating new techniques to better combine several multi-scale branches.

References

[1] Yazan Abu Farha, Alexander Richard, and Juergen Gall. When will you do what? - anticipating temporal occurrences of activities. 06 2018. 2

[2] Guglielmo Camporese, Pasquale Coscia, Antonino Furnari, Giovanni Farinella, and Lamberto Ballan. Knowledge distillation for action anticipation via label smoothing. pages 3312–3319, 01 2021. 2

[3] J. Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. pages 4724–4733, 07 2017. 2

[4] Dima Damen, Hazel Doughty, Giovanni Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. 1

[5] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 4

[6] Jeff Donahue, Lisa Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. Long-term recurrent convolutional networks for visual recognition and description. pages 2625–2634, 06 2015. 2

[7] Chenyou Fan, Jangwon Lee, and Michael Ryoo. Forecasting hand and object locations in future frames. 05 2017. 2

[8] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition, 12 2018. 2

[9] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. 04 2016. 2

[10] Panna Felsen, Pulkit Agrawal, and Jitendra Malik. What will happen next? forecasting player moves in sports videos. pages 3362–3371, 10 2017. 2

[11] A. Furnari, S. Battiato, and G. M. Farinella. Leveraging uncertainty to rethink loss functions and evaluation measures for egocentric action anticipation. In *International Workshop on Egocentric Perception, Interaction and Computing (EPIC) in conjunction with ECCV*, 2018. 1

[12] Antonino Furnari, Sebastiano Battiato, Kristen Grauman, and Giovanni Farinella. Next-active-object prediction from egocentric videos. *Journal of Visual Communication and Image Representation*, 49, 10 2017. 2

[13] Antonino Furnari and Giovanni Farinella. Rolling-unrolling lstms for action anticipation from first-person video. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 1, 2, 3, 4, 5, 6

[14] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. Red: Reinforced encoder-decoder networks for action anticipation. 07 2017. 2

[15] W. Graham, Rob Fergus, Yann Lecun, and Christoph Bregler. Convolutional learning of spatio-temporal features. volume 6316, 12 2010. 2

[16] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. pages 1725–1732, 06 2014. 2

[17] Hema S. Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1):14–29, 2016. 1

[18] Tian Lan, Tsung-Chuan Chen, and Silvio Savarese. A hierarchical representation for future action prediction. volume 8691, pages 689–704, 09 2014. 2

[19] Yin Li, Miao Liu, and James M. Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 639–655, Cham, 2018. Springer International Publishing. 4

[20] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2019. 1

[21] Tahmida Mahmud, Mahmudul Hasan, and Amit Roy-Chowdhury. Joint prediction of activity labels and starting times in untrimmed videos. 10 2017. 2

[22] Jonathan Munro and Dima Damen. Multi-modal Domain Adaptation for Fine-grained Action Recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[23] Seungjun Nah, Tae Kim, and Kyoung Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. pages 257–265, 07 2017. 2

[24] Joe Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. pages 4694–4702, 06 2015. 2

[25] Yulei Niu, Zhiwu Lu, Ji-Rong Wen, Tao Xiang, and Shih-Fu Chang. Multi-modal multi-scale deep learning for large-scale image annotation. *IEEE Transactions on Image Processing*, 28:1720–1731, 04 2019. 2

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 4

[27] Nicholas Rhinehart and Kris Kitani. First-person activity forecasting with online inverse reinforcement learning. 10 2017. 2

[28] Paul Schydlo, Mirko Rakovic, Lorenzo Jamone, and José Santos-Victor. Anticipation in human-robot cooperation: A recurrent neural network approach for multiple action sequences prediction. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, pages 1–6. IEEE, 2018. 1

[29] Fadime Sener, Dipika Singhania, and Angela Yao. *Temporal Aggregate Representations for Long-Range Video Understanding*, pages 154–171. 10 2020. 1, 2, 5

[30] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in Neural Information Processing Systems*, 1, 06 2014. 2

[31] Sijie Song, Jiaying Liu, Yanghao Li, and Zongming Guo. Modality compensation network: Cross-modal adaptation for action recognition. *IEEE Transactions on Image Processing*, 29:3957–3969, 2020. 1

[32] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. pages 4489–4497, 12 2015. 2

[33] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. pages 6450–6459, 06 2018. 2, 5

[34] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. pages 98–106, 06 2016. 1

[35] Xiaohan Wang, Yu Wu, Linchao Zhu, and Yi Yang. Symbiotic attention with privileged information for egocentric action recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):12249–12256, Apr. 2020. 1

[36] Kuo-Hao Zeng, William Shen, De-An Huang, Min Sun, and Juan Carlos Niebles. Visual forecasting by imitating dynamics in natural sequences. 08 2017. 2

[37] Mengmi Zhang, Keng Ma, Joo Lim, Qi Zhao, and Jiashi Feng. Deep future gaze: Gaze anticipation on egocentric videos using adversarial networks. pages 3539–3548, 07 2017. 2