AVT-PAC: A Pipeline For Multimodal Action Prediction and Captioning

Athira Krishnan R[1], *Ambarish Parthasarathy*[1], *Sucharitha Devarakonda*[1], *Lakshmi Sai Rithika Srikurmam*[1], *Swapna M.*[2], *J. V. Satyanarayana*[2], *Sumohana S Channappayya*[1]
[1] *Indian Institute of Technology Hyderabad*
{ai22resch01001@,ai23resch01001@,sucharitha.devarakonda@mae.,rithika.srikurmam@ee.,sumohana@ee.}iith.ac.in
[2]*Research Centre Imarat, DRDO, Hyderabad*
{swapna.rci.drdo, satyanarayana.jv.rci}@gov.in

***Abstract*—Action prediction from frames and videos is a well-studied problem. Models trained with a single modality, mostly vision, will fail in low-light conditions. Recent works have attempted to predict action categories using vision-language and audio-visual models. A challenge, however, is that some dataset annotations lack temporal ground truth and include only the vision modality. Relying on transformers and an intelligent Vision Language Model (VLM) is a viable solution, but deploying them on edge devices could lead to reduced performance and hallucinations. This work presents an Audio-Visual-Text (AVT)-based multi-step Pipeline for Action Prediction and Captioning (AVT-PAC) to address this problem. First, for an input video, we identify the area to focus on using the Region-of-Interest (ROI) Extraction module. CLIP and CLAP encoders are used for ROI prediction. However, the ROI extracted region may vary in duration, resulting in a large number of frames to be processed. To avoid learning from redundant frames, we uniformly sample key frames within the ROI extracted region using a keyframe extraction module. These keyframes are then used to train an Audio-Visual Action and Text-Aware Representation (AVATAR) model to predict actions and captions. Through systematic experiments, we demonstrated that the proposed AVATAR-TCN model beats the present state-of-the-art (SOTA) baselines on the AVE dataset. Code is available in https://github.com/lfovia/AVT-PAC**

***Index Terms*—Multimodal, Audiovisual, Action Prediction, Keyframe extraction, Temporal Localization.**

I. Introduction

Building an intelligent pipeline is essential to solving action prediction problems at scale, especially for autonomous vehicles or surveillance agents, as it is crucial for determining their next actions. Also, the solution built to cater to the needs of such agents should not only perform in real time but also maintain performance on edge computing. In this paper, we tackle one of these issues in the context of multimodal action detection and captioning. The action recognition model aims to address the multiclass classification problem, where the labels correspond to video or action categories. Although this can be solved with a single modality, factors such as occlusion can significantly affect the model's predictions. For a vision-based model, factors such as lighting conditions, adverse weather, and camera movement can significantly impact performance. In such circumstances, we propose relying on audio, a second modality. Further, in certain surveillance scenarios, when an agent or a remote user needs to better understand the scene, captions are helpful. To cater to this need, we also extend a head to predict captions.

This research was supported by Research Centre Imarat (RCI) project RCI/23CR0001/STAR/STAR/CMS-III, DRDO, Government of India.

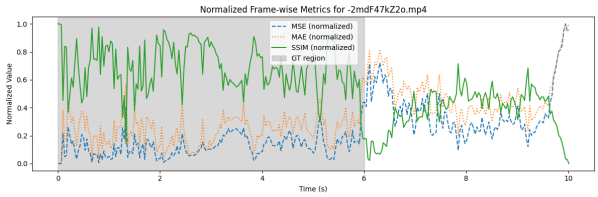

Fig. 1. The comparison of the metrics used on a given video. The plot becomes very noisy when the score is computed between neighboring frames.

In most applications, ROI extraction is guided by an abnormal change in the given scene particularly in detecting the type of actions, and this is solved in [1] using methods that rely on vision metrics like Mean Squared Error (MSE), Mean Average Error (MAE), Structural Similarity Index (SSIM), Contrastive Language-Image Pre-training (CLIP) [2] score, or a dedicated deep learning model [3]. But when computed at the frame level, we can observe that the plots are too noisy, as shown in Fig. 1. Moreover, relying on metrics from a single modality may not be helpful in all scenarios. Inspired by [4], we introduce an audio-based Contrastive Language-Audio Pretraining (CLAP) model [5] to aid the vision counterpart, CLIP, in decision-making.

In our proposed AVT-PAC, we employ a three-in-one pipeline for action recognition and captioning using ROI extraction, keyframe selection, and the AVATAR module to make predictions from a given video sample. The following are the key contributions carried out in this work:

- A multimodal ROI extraction module using CLIP [2] and CLAP [5] models for a given type of activity.
- AVATAR, an audio-visual-text-based multimodal action prediction and captioning model.
- AVT-PAC, an overall three-in-one pipeline for action

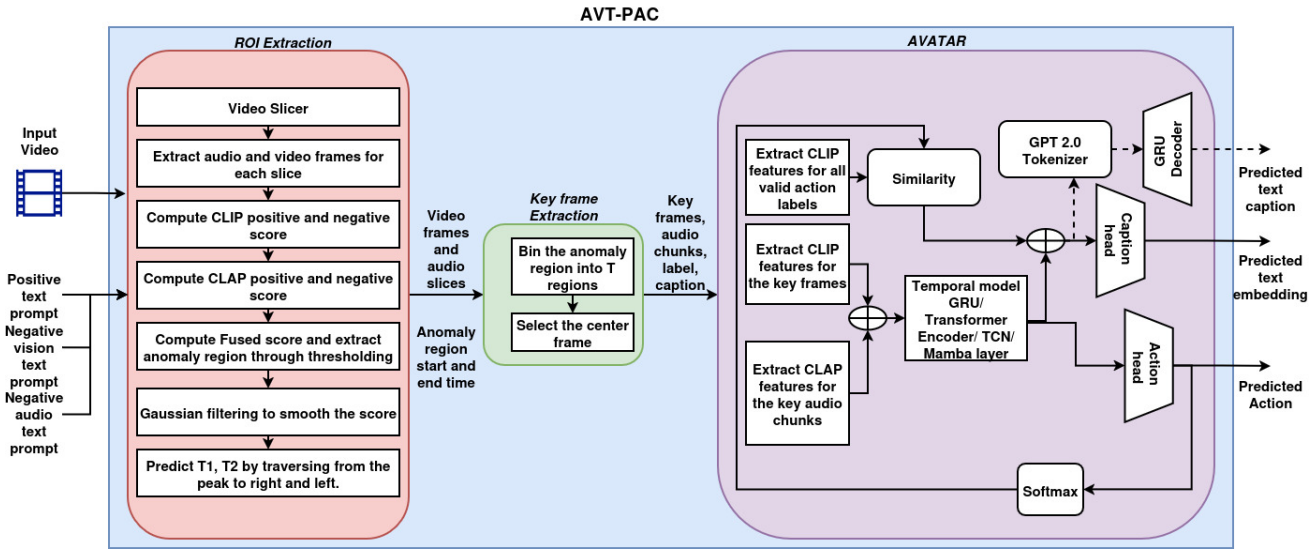

Fig. 2. The proposed three-in-one AVT-PAC framework. It consists of three stages: ROI extraction, key frame extraction, and the AVATAR module.

recognition and captioning, leveraging the capability of ROI extraction and keyframe extraction.

- Validation of the proposed pipeline's real-time feasibility and generalization capability across varied environments.

II. Related Works

Action prediction is a widely researched area that began with vision-based single-modality solutions [6]–[8] and has mostly relied on the spatial and temporal dimensions of visual input. When faced with adversaries or experience occulsions, these solutions struggled to provide reliable predictions. This propelled the works to branch out into audio-visual [9]–[12], multimodal solutions. These works used audio information to maintain action continuity when the subject performing the action moved out of frame, experienced occlusion, or had difficulty visually identifying the subject under unfavorable conditions. Although this improved the reliability of the results, action-label-based action prediction cannot provide a context-based description of the contents of long-form videos or videos with multiple actions that may or may not fall under predefined labels. This is where the text modality provides support through contextual clues and text-based instructions. In the past few years, the exponential growth of Large Language Models (LLMs) has led to the development of Vision-Language Models (VLMs) [13]–[15] that generate short/long free-form text captions for videos/images based on user requirements. This further led to Multimodal Large Language Models (MLLMs) [16]–[18] that tend to support more varied input/output combinations with enhanced logical reasoning and understanding of complex dynamic environments.

Although these works provide improved and robust solutions, they tend to be on the heavier side when it comes to memory and computational requirements, which makes it difficult to provide a relatively lightweight solution that leverages all three modalities: vision, audio, and text, and is sustainable on low-compute devices. Our proposed work attempts to address these constraints and provide a viable lightweight solution.

III. Proposed Approach

Given a short or long-range video from a dataset with only class-level labels, we propose a model that not only localizes the action region via ROI extraction, but also processes the keyframes from the detected region, predicts the action category with a single head, and generates a caption with the other head. The overall architecture of the proposed AVT-PAC pipeline is shown in Fig. 2.

A. ROI extraction module

In the first stage, we propose an ROI extraction approach that predicts the region corresponding to the action in the video using the CLIP [2] encoder f_v and the CLAP [5] encoder f_a. For a dataset $\mathcal{D} = \{V_i, Y_i\}$, V_i is the video sample, ground truth action Y_i, for the i^{th} sample where $|\mathcal{D}| = \mathcal{N}$. The frames are selected every 0.5 seconds as I_j in V_i, and we take an audio sample H_j of length 0.5 seconds. We use competing positive and negative text prompts for each modality and find their cosine similarity (cs). For video, the CLIP [2] confidence score at frame j is:

$$s_j^{\text{clip}} = \mathcal{S}(cs(\mathbf{k}_j,\, f_v(p^+))) - \mathcal{S}(cs(\mathbf{k}_j,\, f_v(p^-))), \quad (1)$$

where p^+ is the positive prompt that describes the action being present, and p^- is the negative prompt that describes an empty background with no activity, and $\mathbf{k}_j = f_v(I_j)$. $\mathcal{S}$ is softmax operator. Similarly, for audio, the CLAP [5] confidence score is:

$$s_j^{\text{clap}} = \mathcal{S}(cs(\mathbf{l}_j,\, f_a(q^+))) - \mathcal{S}(cs(\mathbf{l}_j,\, f_a(q^-))), \quad (2)$$

where q^+ is the positive prompt and q^- is the negative prompt as a silence and $\mathbf{l}_j = f_a(H_j)$. At each timestamp, the modality

with the higher confidence gap $c_j = |s_j|$ is selected, giving a fused score:

$$s_j = \begin{cases} s_j^{\text{clip}} & \text{if } |s_j^{\text{clip}}| \geq |s_j^{\text{clap}}| \\ s_j^{\text{clap}} & \text{otherwise,} \end{cases} \tag{3}$$

A binary prediction $y_j = \mathbf{1}[s_j > 0]$ is computed per frame, denoting whether activity is detected as present in frame j. This is then smoothened using a Gaussian filter and a majority vote window, and merged into contiguous segments, c_j. The timestamp of maximum confidence t^* is located, and the boundaries of the ROI extracted region $[T_1, T_2]$ are found by traversing left and right until confidence drops below $\tau_{\min}$ = 0.05.

$$\begin{aligned} t^* &= \arg\max_j c_j \\ T_1 &= \min\{j \leq t^* \mid c_j \geq \tau_{\min}\}, \\ T_2 &= \max\{j \geq t^* \mid c_j \geq \tau_{\min}\} \end{aligned} \tag{4}$$

B. Keyframe extraction module

If the ROI extracted region $[T_1, T_2]$ is predicted for 3s, then for a 30 fps video, that results in 90 frames to be processed. In order to remove or neglect any duplicate or redundant frames, we bin the ROI extracted region into $\mathcal{K}$ regions and select the center frame as the keyframe. This now represents the ROI extracted region as $(vid_{1...\mathcal{K}}, aud_{1...\mathcal{K}})$, where $vid_{1...\mathcal{K}}$ is $\mathcal{K}$ video keyframes, $aud_{1...\mathcal{K}}$ is $\mathcal{K}$ audio slices centered around the video keyframe timestamp. We have empirically chosen $\mathcal{K}$ value as 16.

C. AVATAR Module

Now from the prior keyframe extraction, we curate a dataset with $\mathcal{N}$ samples as $\mathcal{D}_2 = \{vid^i_{1...K}, aud^i_{1...K}, y^i_{act}, y^i_{cap}\}$, where i varies from 1 to $\mathcal{N}$, y^i_{act} is ground truth action Y_i, and y^i_{cap} is the ground truth caption obtained from CLAP [5] model for the i^{th} sample, which will be used to train an AVATAR model to predict actions and caption embedding.

The video keyframes $vid_{1...K}$ are passed through the vision encoder f_v. Audio chunks $aud_{1...K}$ are also passed through the audio encoder f_a. The encoded data is then concatenated and passed to the temporal modeling block. Further, the learned representations are passed through two MLP heads to predict the corresponding action and caption. In case of the caption, given a list of classes, CLIP [2] embeds the labels to compute similarity using the softmax of the predicted action logits. This similarity helps identify the predicted class, which is then concatenated with the learned temporal audio-visual embedding and passed to the caption head to predict the caption.

$$F_\theta = \text{MLP}(\text{Pool}(\text{Temporal}([f_v(\mathbf{v}_{1:K});\, f_a(\mathbf{a}_{1:K})]))) \tag{5}$$

To ensure learning, we use three loss components, namely the Action Loss, the Contrastive Loss, and the Caption Loss. $\mathcal{L}_{act}$ is the cross-entropy loss between the predicted and ground truth action.

$$\mathcal{L}_{\text{act}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y^{i,c}_{act} \log \hat{y}^{i,c}_{act} \tag{6}$$

where $y^{i,c}_{act}$ are the one-hot pseudo-labels, and $\hat{y}^{i,c}_{act}$ are the predicted class probabilities for the i-th sample belonging to class c. Using a temperature coefficient τ, the $\mathcal{L}_{cont}$ measures the contrastive loss between audio and video features.

$$\mathcal{L}_{\text{cont}} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log\frac{e^{\mathbf{k}_i^\top \mathbf{l}_i/\tau}}{\sum_j e^{\mathbf{k}_i^\top \mathbf{l}_j/\tau}} + \log\frac{e^{\mathbf{l}_i^\top \mathbf{k}_i/\tau}}{\sum_j e^{\mathbf{l}_i^\top \mathbf{k}_j/\tau}}\right]. \tag{7}$$

Similarly, $\mathcal{L}_{text}$ is the loss between the predicted $\hat{y_t}$ and ground truth text embedding y_t for probable classes using CLIP as shown in Equation 8.

$$\mathcal{L}_{text} = 1 - cs(\hat{y_t}, y_t). \tag{8}$$

Total loss sums up as,

$$\mathcal{L}_E = w_1\mathcal{L}_{\text{act}} + w_2\mathcal{L}_{\text{cont}} + w_3\mathcal{L}_{\text{text}}, \tag{9}$$

where w_1 , w_2, w_3 are the scalar weights given to each loss term, selected empirically as 1, 0.5, and 0.3, respectively. Finally, the model parameters are learned according to

$$\theta_1^* = \arg\min_\theta \mathcal{L}_E. \tag{10}$$

Through this formulation, we get the predicted class and the text embedding. Given a set of $\mathcal{P}$ valid captions where $\mathcal{P} \ll \mathcal{N}$, by computing the cosine similarity of the predicted text embedding to captions in $\mathcal{P}$, we can retrieve the closest text caption. But to predict the text itself, we need a decoder in the caption head. We have tried using a GRU-based decoder with GPT-2.0 tokenizer to learn the caption through $\mathcal{L}_{cap}$,

$$\mathcal{L}_{\text{cap}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\log\left(\frac{\exp\left(\mathbf{z}_{i,k,y_{i,k}}\right)}{\sum_{t=1}^{V}\exp\left(\mathbf{z}_{i,k,t}\right)}\right) \tag{11}$$

where $\mathbf{z}_{i,k,t}$ denotes the predicted logit for the t-th vocabulary token at time step k, and V is the vocabulary size. The total loss is given by

$$\mathcal{L}_D = \mathcal{L}_{\text{cap}} + \lambda\mathcal{L}_{\text{act}}. \tag{12}$$

Where $\lambda = 0.5$ is an empirically chosen regularization coefficient to emphasize decoder-based caption learning, with action prediction initialized from the no-decoder model. The model parameters are optimized according to,

$$\theta_2^* = \arg\min_\theta \mathcal{L}_D. \tag{13}$$

IV. Results and Discussion

In our proposed pipeline strategy, different versions of AVATAR modules are compared with the baseline and SOTA approaches based on their respective computational complexities.

A. Implementation Details

We have done all our training on the AVE dataset. As the AVE dataset lacks ground truth text captions, we have relied on the msclap [5] model to curate the same. The models were trained for 15 epochs with a batch size of 16 on a NVIDIA RTX 3090 workstation with 24 GB of GPU. The AVE dataset is split into an 80:20 training-validation ratio, with 3,277 training samples and 820 validation samples.

TABLE I

QUANTITATIVE COMPARISON OF ACTION PREDICTION. THESE MODELS WERE TRAINED ON THE AVE DATASET [19]. THE **BOLD** ONE DENOTES THE BEST PERFORMING MODEL.* THE CORRESPONDING MODEL SIZE WAS NOT MENTIONED IN [20]

Models	Dataset	Modality	Model Size	Accuracy	Caption Head	Cosine Similarity	CIDEr	ROUGE-L
Audio guided Visual attention [19]	AVE	AV	1.33M	0.740	-	-	-	-
DMRN [19]	AVE	AV	1.57M	0.732	-	-	-	-
AGSP-DSA [20]	AVE	AVT	*	0.934	-	-	-	-
AVATAR-TCN	AVE	V	3.42M	0.8829	Embedding	0.9281	-	-
AVATAR-TCN	AVE	A	4.47M	0.8171	Embedding	0.9306	-	-
AVATAR-TCN Decision level ensemble	AVE	AV	7.89M	0.8720	Embedding	0.9303	-	-
AVATAR-GRU	AVE	AVT	3.42M	0.9060	Embedding	0.9333	-	-
AVATAR-Transformer	AVE	AVT	8.42M	0.9146	Embedding	0.9298	-	-
AVATAR-TCN	AVE	AVT	4.47M	**0.9463**	Embedding	0.9310	-	-
AVATAR-Mamba	AVE	AVT	3.81M	0.9378	Embedding	**0.9336**	-	-
AVATAR-TCN-Mamba	AVE	AVT+Decoder	31.69M	0.9317	Mamba Decoder	-	1.5783	0.5224
AVATAR-TCN-GRU	AVE	AVT+Decoder	57.56M	0.9439	GRU Decoder	-	**1.8859**	**0.5948**
AVATAR-TCN-GRU	HMDB51	AVT+Decoder	57.56M	0.7455	GRU Decoder	-	0.0498	0.2246

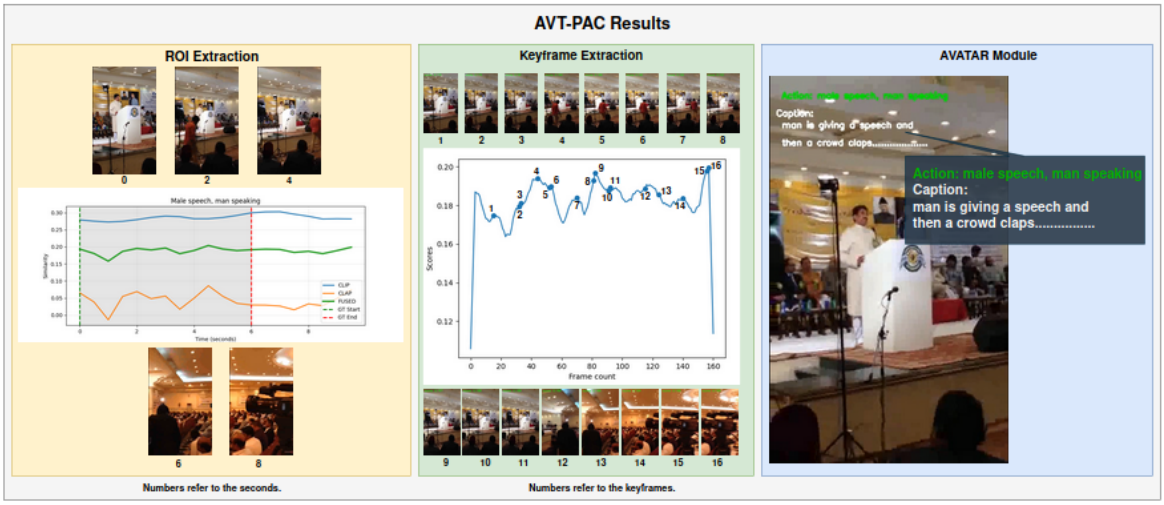

Fig. 3. The qualitative results of the three-in-one AVT-PAC Framework. The ROI extraction (left), Key frame extraction (centre), and the proposed AVATAR module (right) for a given video sample.

B. Dataset quality analysis

To ensure the quality of CLAP-generated labels [5] for training, we have curated captions for the validation fold of AVE using a human annotator. We have computed the inter-dataset agreement for the validation fold using the CiDEr [21] and ROUGE [22] scores. Despite having only 1 groundtruth variant, the CiDEr score of *0.3469* and ROUGE score of *0.3012* are reasonably good, indicating a weak but acceptable overlap between the generated caption and the groundtruth.

C. Results Analysis

Quantitative results for the experiments done for the AVATAR module are tabulated in Table I. For the action recognition part, it can be observed that the sequential models significantly outperform the previous Audio guided vision attention models and the DMRN [19] in terms of accuracy. With AVT modalities leveraging a graph-based model, [20] is the current SOTA on the AVE dataset. It is interesting to note that the TCN-based AVATAR model outperforms the other variants, which is approximately 50% lighter than the Transformer encoder-based variant. In captioning, although the Mamba model [23] performs better, the delta of improvement is small. For decoder-based variants of AVATAR-TCN, we find that the GRU variant outperforms the Mamba variant.

Further, to test the generalization capability, we use the HMDB51 dataset [24] without audio. With the frozen decoder layer, our model consistently performs well on the HMDB51 dataset. Similarly, to test the model's capability to generalize and work in the absence of the audio modality, we have tried zero-shot inferencing on a few samples from the MUSIC21 [25], UCF Crime dataset [26], and HMDB51 dataset, as shown in Fig. 5.

To understand the model behaviour, we visualize t-SNE plots as shown in Fig. 4. In AVATAR modules with TCN and Mamba-based models, the clusters are more closely packed. The mamba-based clusters are relatively spread out, which could result in false predictions. The qualitative analysis on a given video sample is shown in Fig. 3. In the case of the ROI extraction (left), the variation of CLIP [2], CLAP [5], and the fused score can be analysed. Similarly, for the extracted keyframe (middle), the selected 16 keyframes from the 160 frames are shown here. Lastly, results from the AVATAR module (right), shows the predicted action, and caption. To achieve real-time performance testing, we compared the inference time of the AVATAR-TCN-GRU model for a video sample on constrained computes. Inference times of 0.33s and 1.43s were

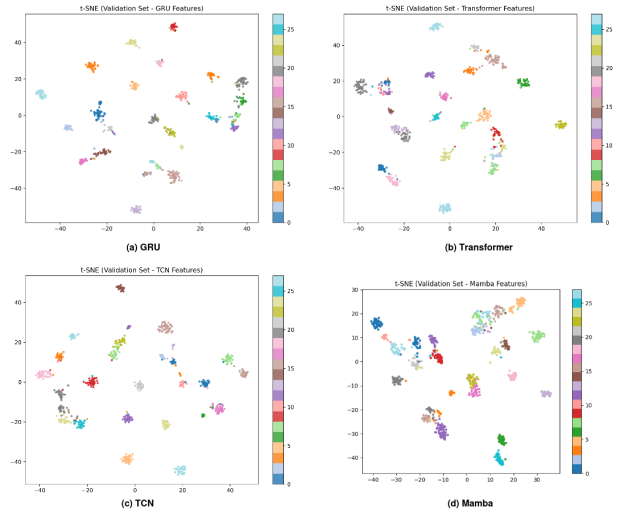

Fig. 4. The corresponding t-SNE plots of the trained models. It is observed that the TCN model has better, compact, and well-separated clusters.

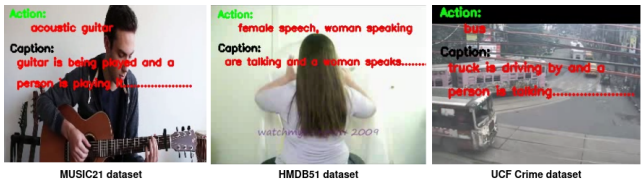

Fig. 5. Zero-shot performance of models trained on the AVE dataset on a sample taken from the MUSIC21, HMDB51, and UCF-Crime datasets.

achieved on a NVIDIA RTX 3090 24GB workstation and a NVIDIA TITAN Xp 12GB GPU workstation, respectively.

V. Conclusion

In this paper, we introduce AVT-PAC, an end-to-end three-in-one pipeline that handles ROI extraction, keyframe extraction, and the proposed multimodal audio-visual-text framework, AVATAR. AVATAR takes inputs from keyframe extraction, predicts the action category, and generates captions for the given sequence of frames. It was observed that the TCN-based and the Mamba-based AVATAR performed consistently better, providing a good balance between performance and computation. Additionally, the TCN model with a GRU decoder consistently performed better across all evaluations. Furthermore, there is a good improvement as compared to the SOTA [20] as well as the baselines [19] on the AVE dataset. The future directions for this work could include generating more robust multimodal ground-truth captions, optimizing the model, and learning more effective attention-based representations.

References

[1] E. Schwartz *et al.*, "Maeday: Mae for few- and zero-shot anomaly-detection," *Computer Vision and Image Understanding*, vol. 241, p. 103958, 2024.

[2] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," pp. 8748–8763, PmLR, 2021.

[3] A. Alshalawi *et al.*, "Fire swin: Video anomaly detection using a hybrid model with convolutional layers, fire module, and swin transformer," *Journal of Engineering Research*, vol. 14, no. 1, pp. 811–819, 2026.

[4] A. K. R *et al.*, "Av-io: Audio-visual indoor/outdoor classification dataset via multi-modal weak supervision," in *2026 National Conference on Communications (NCC) In Press*, IEEE, 2026.

[5] B. Elizalde *et al.*, "Clap learning audio concepts from natural language supervision," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, IEEE, 2023.

[6] A. Arnab *et al.*, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6836–6846, 2021.

[7] A. Rasouli, "Deep learning for vision-based prediction: A survey," *arXiv preprint arXiv:2007.00095*, 2020.

[8] P. Schydlo *et al.*, "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)*, pp. 5909–5914, 2018.

[9] C. Wang *et al.*, "Exploring multimodal video representation for action recognition," in *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 1924–1931, 2016.

[10] Y. Wei *et al.*, "Learning in audio-visual context: A review, analysis, and new perspective," 2022.

[11] R. Gao *et al.*, "Listen to look: Action recognition by previewing audio," 2020.

[12] J. Chalk *et al.*, "Tim: A time interval machine for audio-visual action recognition," 2024.

[13] J.-B. Alayrac *et al.*, "Flamingo: a visual language model for few-shot learning," *Advances in neural information processing systems*, vol. 35, pp. 23716–23736, 2022.

[14] Y. Li *et al.*, "Llama-vid: An image is worth 2 tokens in large language models," in *European Conference on Computer Vision*, pp. 323–340, Springer, 2024.

[15] M. Maaz *et al.*, "Video-chatgpt: Towards detailed video understanding via large vision and language models," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12585–12602, 2024.

[16] K. Li *et al.*, "Mvbench: A comprehensive multi-modal video understanding benchmark," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22195–22206, 2024.

[17] X. Li *et al.*, "Videochat-flash: Hierarchical compression for long-context video modeling," *arXiv preprint arXiv:2501.00574*, 2024.

[18] J. Zhu *et al.*, "Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models," *arXiv preprint arXiv:2504.10479*, 2025.

[19] Y. Tian *et al.*, "Audio-visual event localization in unconstrained videos," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 247–263, 2018.

[20] K. Karthikeya *et al.*, "Adaptive graph signal processing for robust multimodal fusion with dynamic semantic alignment," *Scientific Reports*, 2026.

[21] R. Vedantam *et al.*, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.

[22] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, (Barcelona, Spain), pp. 74–81, Association for Computational Linguistics, July 2004.

[23] A. Gu *et al.*, "Mamba: Linear-time sequence modeling with selective state spaces," in *First conference on language modeling*, 2024.

[24] H. Kuehne *et al.*, "Hmdb: a large video database for human motion recognition," in *2011 International conference on computer vision*, pp. 2556–2563, IEEE, 2011.

[25] H. Zhao *et al.*, "The sound of pixels," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 570–586, 2018.

[26] W. Sultani *et al.*, "Real-world anomaly detection in surveillance videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6479–6488, 2018.