2nd AVA Challenge@IEEE MIPR 2024 Team: ServerDown

Tahsen Islam Sajon
Machine Learning Engineer
ACI Limited
Dhaka, Bangladesh
sajon.tahsen@gmail.com

Sabbir Hossain Ujjal
Machine Learning Engineer
ACI Limited
Dhaka, Bangladesh
sabbirhossainujjal.buet@gmail.com

Hasan Zohirul Islam
Machine Learning Engineer
ACI Limited
Dhaka, Bangladesh
hzihimel@gmail.com

***Abstract*—This report details our approaches and experiments for the 2nd AVA Challenge at IEEE MIPR 2019. We explored various methods, including Per-Frame Classification with CNNs, Pretrained CNN Encoder with RNNs, and state-of-the-art transformer-based models VidSwin and ViVit. Our most effective method integrated EfficientNetV2M with a Transformer block, enabling end-to-end training and improved spatial-temporal feature learning. Finally, an ensemble of the CNN-Transformer model and the Per-Frame Classification approach enhanced generalization and accuracy, providing a robust solution for accident risk prediction. These approaches provided valuable insights into the nature of the problem and established baselines. Code and other relevant materials are available on our *github repository*.**

I. Approaches

A. Per-Frame Classification with CNN

In our first approach, we performed frame-level classification using Convolutional Neural Networks (CNNs). We chose to use the last frames of a video rather than all frames, based on the observation that the final frames are more likely to contain relevant indicators of risk, while training on all frames could result in noisy labels due to the inclusion of irrelevant frames. We experimented with using the last 1, 3, and 5 frames of a video. We averaged the prediction values to obtain the final output when using more than one frame.

As our feature extractors, we selected state-of-the-art (SOTA) architectures that provide distinct mechanisms to model image features effectively. These architectures included:

- **EfficientNetV2 and EfficientNetB5**: Known for their efficient scaling and high performance on image classification tasks [1].
- **ConvNext**: A modernized version of the standard ResNet architecture with improved performance [2].
- **ResNext**: Utilizes Squeeze and Excite blocks, which recalibrate feature responses by explicitly modelling channel interdependencies [3].

This method aimed to establish a robust baseline, as well as determining a suitable architecture to extract the frame features. We used batch normalization and L2 regularization, as well as dropout, to prevent overfitting and improve generalization.

B. Pretrained CNN Encoder with RNN

In our second approach, we used a pretrained CNN as an encoder to extract image feature representations, which were then passed to a Recurrent Neural Network (RNN). This method allowed the RNN to leverage the temporal information present in the sequences of frames. Initially, we used ImageNet pretrained models, but later opted to use models trained from our single-frame approach, believing that they had already learned some relevant features. By adding the RNN, we aimed to enhance the robustness of the classifier by capturing temporal dependencies. For this approach, we experimented with Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks, processing up to 128 frames. This approach, however, did not yield improvements.

C. State-of-the-Art Video Classification Models

In our third approach, we aimed to leverage the capabilities of transformers for video classification tasks. Transformers have shown superior performance in capturing temporal dynamics and modeling long-range dependencies in video sequences. We selected two state-of-the-art transformer-based models: VidSwin and ViVit.

- **VidSwin**: A variant of the Swin Transformer adapted for video classification [4].
- **ViVit**: A video vision transformer that processes video frames as tokens to models their interactions effectively [5].

We used sequences of 16 and 24 frames as input for these models. These models were trained with the same preprocessing techniques applied to the video sequences, ensuring consistency in the input data.

D. End-to-End CNN-Transformer Model

Our initial attempts with the CNN+RNN approach revealed limitations in capturing temporal dependencies and did not outperform the single-frame baseline. To address this, we chose to integrate the CNN as an encoder with a Transformer block within the computation graph to better learn relevant spatial features and effectively model temporal dependencies.

We used the EfficientNetV2 model, which performed best in our single-frame approach, as the image feature encoder. The transformer block, with 256 feature dimensions, was employed

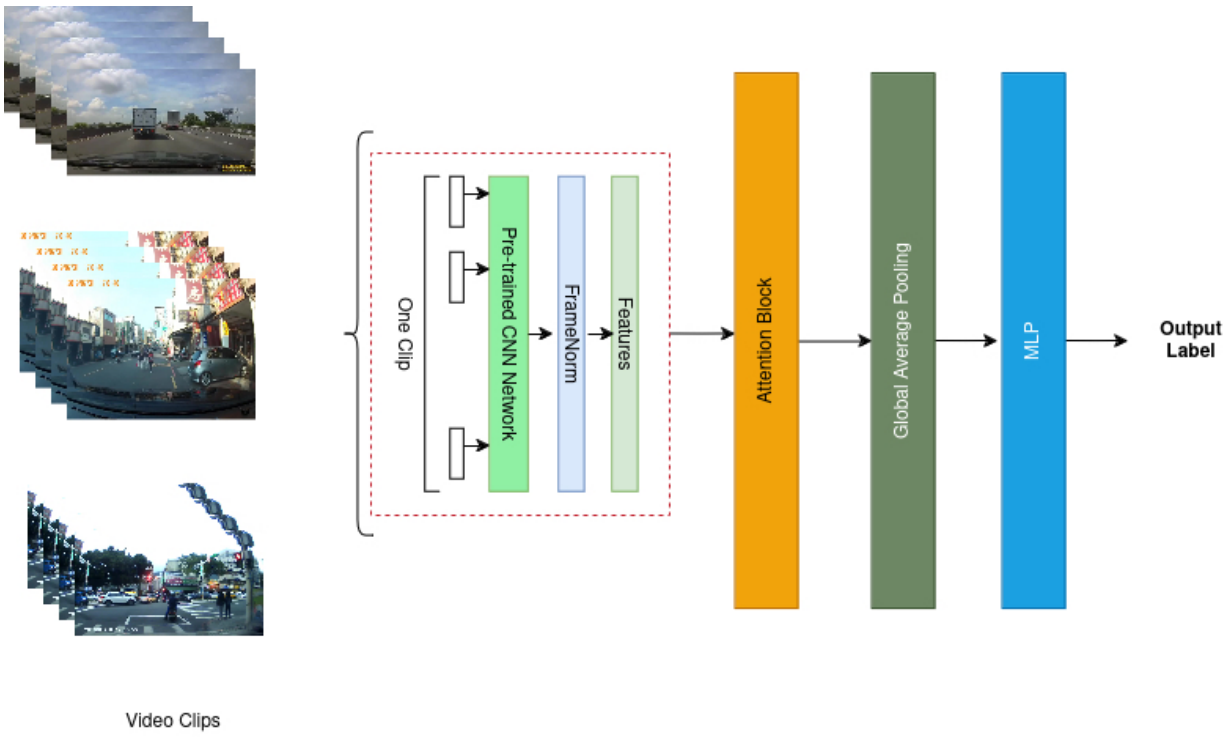

Fig. 1. Our End-to-End CNN-Transformer Architecture

to model the temporal dependencies. A dense layer was used to project the 1280-dimensional image features into the 256-dimensional transformer space, allowing the model to focus on the most relevant aspects of the input data while keeping the computation requirements lower. An MLP (Multi-Layer Perceptron) on the transformer outputs was used to generate the final predictions.

This end-to-end training setup improved the model's ability to capture both spatial and temporal dependencies, providing a robust solution for predicting accident risk from video sequences.

*E. Ensemble Method **(Proposed)***

To leverage the strengths of different models and improve overall performance, we employed an ensemble method. Our ensemble combined predictions from the End-to-End CNN-Transformer Model and the Per-Frame Classification with CNN approach.

The rationale behind this ensemble strategy was to utilize the robust feature extraction capabilities of the single-frame CNN models alongside the advanced temporal modeling of the CNN-Transformer model. By averaging the softmax outputs of these models, we aimed to enhance the generalization ability and predictive accuracy.

Specifically, the ensemble method involved the following steps:

1) **Training Individual Models**: Both the Per-Frame Classification with CNN and End-to-End CNN-Transformer models were trained independently using their respective training procedures.
2) **Generating Predictions**: For each video, both models produced probability scores for the accident risk class.
3) **Averaging Predictions**: The softmax outputs from the two models were averaged to obtain the final prediction.

II. Experiment Results

In Table I, we highlight our experiment results across the different approaches. Detailed implementation, including code, is provided in our GitHub repository.

III. Challenges and Future Scope

A. Challenges

In the context of dashcam-based accident prediction, a significant challenge arose when an accident captured by the dashcam occurred outside the immediate path of the recording vehicle. This scenario often led to confusion for the predictive model, as the incident could be classified as risky despite posing no immediate threat to the vehicle itself. Such instances resulted in misclassification, where the model predicted risk based on the occurrence of an accident visible in the camera frame but not directly impacting the vehicle's trajectory. Addressing this challenge required refining the model's understanding of spatial context and trajectory relevance, ensuring that predictions accurately reflected the actual risk to the recording vehicle rather than solely reacting to visible incidents.

TABLE I
EXPERIMENT RESULTS

Experiment	Public ROC	Private ROC
Per-Frame Considering Last Three Frames	0.6873	0.7428
VidSwin	0.6619	0.7118
Ensemble (Average) (CNN-Transformer and Per Frame)	0.7459	0.7005
End-to-End CNN Transformer	0.6905	0.6820
Ensemble (Weighted) (CNN-Transformer and Per Frame) (Selected Submission on Kaggle)	0.7560	0.6576
Pretrained-CNN + RNN	0.6280	0.6571

An important challenge we faced was the variability in weather, lighting, and road conditions, which introduced complexities affecting the reliability of predictive models. To address this issue, we implemented strategies using data augmentation techniques to account for diverse environmental scenarios. By incorporating a wide range of driving conditions in our training data, we aimed to enhance the model's robustness and its capacity to generalize effectively across various real-world scenarios.

Additionally, due to the limited amount of data and the complexity of larger models, overfitting issues were encountered. To mitigate this challenge, common techniques such as dropout, weight decay, early stopping, and cross-validation were applied. These methods were essential in fine-tuning the models to ensure they are generalized effectively across different datasets and real-world conditions.

B. Future Scope

In our future research, we aim to enhance accident risk prediction in video data by leveraging Graph Neural Networks (GNNs). Our approach involves detecting and tracking objects such as cars and pedestrians in each frame, extracting their bounding box coordinates, and analyzing the relative distances between these objects and the recording vehicle (dashcam). We hypothesize that sudden and significant changes in these relative distances could indicate potential accident scenarios. By applying GNNs, we plan to model the spatial and temporal dependencies within these object interactions, enabling more accurate and proactive accident risk assessments. This methodological advancement not only aims to improve the real-time detection capabilities but also to enhance overall safety and efficiency in autonomous driving and traffic management systems.

We also plan to employ visualization tools akin to Grad-CAM for images. This will enable us to analyze which frames and temporal segments the model emphasizes when making risk predictions. By visualizing these attention patterns, we aim to gain insights into the decision-making process of the model and validate its reasoning behind risk assessments. This approach not only enhances interpretability but also facilitates potential improvements in model robustness and accuracy by identifying critical moments in video sequences that contribute to risk prediction.

Additionally, there exists potential to explore self-supervised learning techniques tailored for video data, such as contrastive learning and video inpainting, aimed at enhancing the model's ability to generalize across diverse video contexts. The investigation of generative models such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) for video synthesis and augmentation could significantly augment training data and bolster model robustness. Further development of methodologies for domain adaptation, including techniques like domain adversarial training (DAT) and meta-learning approaches, holds promise for facilitating knowledge transfer across various video domains and environmental conditions. Moreover, exploring advanced temporal reasoning models such as Temporal Convolutional Networks (TCNs), Recurrent Neural Networks (RNNs) with attention mechanisms, Temporal Relational Networks (TRNs), and Spatio-Temporal Graph Convolutional Networks (ST-GCNs) could enhance predictive accuracy and robustness in video analysis tasks by effectively capturing long-term dependencies and causal relationships within video sequences.

IV. CONCLUSION

In this study, we investigated deep learning methodologies for dashcam-based accident prediction, aiming to enhance model accuracy and robustness in real-world driving scenarios. Our experiments highlighted the efficacy of advanced temporal reasoning models and attention mechanisms in effectively identifying potential accident risks. Despite challenges posed by variability in driving conditions and incidents occurring off the vehicle's path, our models exhibited promising capabilities in predictive accuracy. Looking forward, future research could explore novel techniques like self-supervised learning and generative models to augment and synthesize video data, addressing ongoing challenges such as data scarcity and model interpretability. These advancements are crucial for further improving the reliability and practical application of predictive models in real-world accident prevention and autonomous driving systems.

REFERENCES

[1] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*. PMLR, 2021, pp. 10 096–10 106.

[2] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 11 976–11 986.

[3] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[4] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video swin transformer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 3202–3211.

[5] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6836–6846.