Automatic Soccer Game Highlight Detection

Fang Shu
Department of MS&E
Stanford University
fangshu@stanford.edu

Haoxiang "Mike" Yang
Department of Computer Science
Stanford University
yanghx@stanford.edu

Abstract

Generation of post-game highlights is essential for engaging sports fans and enhancing the viewer experience. This project aims to develop an automated system for detecting highlight-worthy moments in soccer game videos. We collected a dataset comprising 25 full 90-minute soccer match videos, which we segmented into 5-second clips. Each clip was then classified as either containing a highlight (1) or not (0) using a combination of Convolutional Neural Networks and Transformer models. We employed data augmentation, undersampling, and hyperparameter tuning to enhance model performance. The best model achieved a recall of 0.56. Despite these efforts, achieving high performance proved challenging due to the inherent complexity of soccer gameplay. Subtle actions, such as counterattacks, dangerous passes, and high pressing, were difficult for the models to identify as significant events, leading to suboptimal results. Nevertheless, this project underscores the potential of computer vision techniques in automating sports highlight generation, offering a scalable solution for efficient video analysis and content creation.

1. Introduction

Post-game highlights are essential for engaging sports fans. In soccer, highlight production varies by broadcaster and tournament. For example, CBS Sports uploads 15-minute UEFA Champions League highlights immediately after the matches conclude, while Italian Serie A releases 3-minute highlights only hours later. Notably, for most sports leagues, the majority of highlight production still relies heavily on manual editing processes. Our project aims to automate the highlight selection process, improving efficiency and accessibility for tournaments operating with limited resources. By automating this process, we can ensure quicker turnaround times for highlight production and provide consistent, high-quality content across different leagues and tournaments.

In this project, we aim to identify highlight-worthy clips from a full soccer game replay. Our dataset comprises 25 full-length soccer game videos, each spanning 90 minutes. To facilitate our analysis, we segment these videos into 5-second clips and carefully label each clip as either "highlight-worthy" (assigned a label of 1) or "not highlight-worthy" (assigned a label of 0). The detailed procedure for data labeling is elucidated in the Dataset section. Formally, we frame the task as a binary video classification problem: given a 5-second clip extracted from a soccer game, our objective is to predict whether it qualifies as a highlight-worthy moment. To achieve this, we employ state-of-the-art Convolutional Neural Network (CNN) and Transformer models, leveraging their respective strengths in capturing spatial features and long-range dependencies within video data.

2. Related Work

[1] is the most relevant study to our topic of highlight generation. It employs models including Faster R-CNN and Yolo to identify important events in soccer matches. Then based on the key events, the models extract highlights by capturing a few seconds before and after the key events. While the final highlights catch mostly the key moments of the games, a major limitation of the paper is relying solely on object detection on individual frames, rather than incorporating sequences of images (i.e. video clips) to identify the actual dynamics in a soccer match.

[7] analyzes 5-second video clips of soccer games to detect whether the action leads to a goal. The study uses transfer learning to fine-tune an Inflated 3D Networks model. The approach of using video clips instead of images inspires our methodology. However, [7]'s data only allows for goal detection. A soccer game highlight should capture not only goals, but important moments such as close misses, red cards, and penalties. In our study, we aim to incorporate such key moments as well in the highlights.

[5] generates highlights for e-sports, specifically, League of Legends. It uses real-time in-game statistics to calculate the win-loss probability at any given time, and defines high-

lights as moments when the change rate of win-loss probability is large. This innovative approach is not suitable for soccer due to the distinct nature of in-game statistics. League of Legends has a large amount of quantitative data such as kills, coins, and dies that directly relate to the win-loss probability. This is not the case for a soccer game. For instance, a team with higher possession would not necessarily win. However, we draw inspirations from the architecture of this study, in particular, the CNN models used for feature extractions.

[11] proposes alternative approach to understand game dynamics based on the live text captions. This methodology demonstrates success for learning micro-events. However, the proposed model trains on Chinese text caption data that cannot easily adapt for game captions in other languages. This limitation is especially a concern if we want to generate highlights for lower-tier regional tournaments. Given we do not have access to high-quality text caption data in multiple other languages, we decided to focus on directly analyzing video data.

[6] uses a probabilistic Bayesian belief network based on observed events to assign semantic concept-labels to the exciting clips, such as goals, saves, yellow-cards, red-cards, and kicks in soccer video sequences. The labeled clips are then selected according to their degree of importance to include in the highlights. While this approach was capable of capturing the aforementioned key events, it fails to capture other important actions, including dangerous passes, counterattacks, etc. Additionally, it uses audio features for clip selection, while our work relies solely on video/image data.

[3] proposes an approach of unsupervised learning based on the analysis of spatio-temporal local features of video frames. It explores the local visual content of video frames by focusing on spatial and temporal learned features in a low-dimensional transformed sparse space. While the study results have shown promising findings, we deem this study to be less suitable for our project since our main focus is to use computer vision approaches to perform our task.

[13] begins with an improved Shot Boundary Detection technique to accurately identify shot changes. They classify the detected shots into two distinct view types and implement a template-based approach to detect replays within each shot. The system then generates play-break sequences through a rule-based method. The extracted features from these play-break sequences are subsequently input into a multi-kernel Support Vector Machine classifier, which effectively discerns various events. A major limitation of this approach is that the reliance on rule-based methods might limit adaptability to different types of soccer matches or other sports. Additionally, this study does not explicitly incorporate temporal dynamics, which are crucial for understanding the sequence of events in a video.

The authors of [9] conducted a thorough search of diverse action-spotting models to determine the optimal backbone for their highlight classifier. They selected NetVLAD++ as the backbone model. By combining NetVLAD++ with their highlight classifier, they developed a system capable of producing natural and coherent highlight reels. While the authors claimed their approach generated high-quality highlights, they did not compare their highlights with the ground truth highlight videos, which makes it hard to compare their model performance in terms of numerical metrics.

[8] used a multimodal approach by combining video and audio models through both early and late fusion techniques. Results indicated that combining multiple modalities generally improved event detection performance, with significant gains for goal detection due to the distinctive audio cues associated with goals. However, the benefits of multimodal approaches were less pronounced for detecting cards and substitutions, suggesting that different event types might benefit differently from multimodal data. Nevertheless, this study sheds light on our future work, where we could explore directions such as event-specific multimodal strategies and exploration of additional event categories.

The paper [4] introduces a method to temporally locate highlights in sports events by analyzing audience behavior, utilizing a deep 3D CNN on cuboid video samples. The model discerns different levels of spectator excitement and employs a spatial accumulator to generate a score indicating the likelihood of an interesting highlight at a given time. The study uses audience behavior as a proxy for detecting highlights, which is an innovative and less explored method. On the other hand, this approach only showed good result on a hockey dataset, and therefore its performance remains unclear on longer and more dynamic soccer game data.

3. Methods

3.1. Models

We use the *Single-Frame 2D CNN* model as our baseline. We note that 2D CNNs are highly effective for tasks that involve spatial data such as images. They are capable of automatically learning and extracting hierarchical features from raw pixel data, which makes them suitable for a wide range of image-related tasks, including classification, detection, and segmentation. However, 2D CNNs process each frame (image) independently, which makes them less suitable for tasks where temporal dynamics are crucial, such as video analysis. They do not capture dependencies across time, which is essential for understanding sequences of frames. To adapt a Single-Frame 2D CNN to the video classification task, we first classify the data at the frame level. Then at test time, for a clip that consists of multiple frames, if any one of the frames has a predicted label of 1, we assign all frames in this clip labels of 1. Our reasoning is that if any

moment is significant and highlight-worthy, then the immediate leading and trailing actions should also be included in the highlights for context. Figure 1 is a diagram that represents our approach. Table 1 lays out our 2D CNN model architecture.

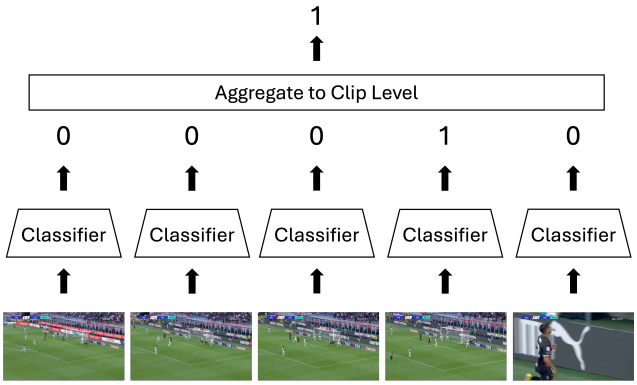

Figure 1. Aggregate frame-level classification to clip level

Layer	Configuration
Conv2D (3 x 3, 3 $\rightarrow$ 16)	stride = 2, padding = 1
BatchNorm2D	
Conv2D (3 x 3, 16 $\rightarrow$ 32)	stride = 2, padding = 1
BatchNorm2D	
Conv2D (3 x 3, 32 $\rightarrow$ 64)	stride = 2, padding = 1
BatchNorm2D	
GlobalAvgPool2D	
Linear	

Table 1. 2D-CNN Model Architecture

Another model we employ is *3D CNN*. The 3D CNNs are more advanced than 2D CNNs in handling video data primarily because they can capture both spatial and temporal features simultaneously. This joint processing enables the model to extract features that are not only spatially relevant but also temporally coherent, enhancing its ability to recognize complex patterns and movements in video data such as soccer matches. The lower layers capture low-level visual patterns such as edges or textures and short-term temporal dynamics, while higher layers learn more abstract and complex concepts such as object parts or player actions along with long-term temporal dependencies. Table 2 lays out our 3D CNN model architecture.

We train the two aforementioned models from scratch. To evaluate the performances of self-trained models versus finetuned models, we also analyzed two pretrained models.

One pretrained model we utilized is the *Vision Transformer (ViT)* . Instead of using convolutions like traditional neural networks, ViT [2] splits an image into small patches and treats each patch as a token. These tokens are then processed through transformer layers, which effectively capture the relationships between different parts of the image. By doing so, ViT can learn complex patterns and dependencies in visual data. In this project, we use the ViT model pretrained on ImageNet-21k at resolution 224x224 from HuggingFace [12]. We then finetune the last layer of the ViT on our data. Similar to Single-Frame 2D CNN, since ViT classifies data at the frame level, if any of the five frames has a predicted label of 1 at test time, we assign all frames in the clip labels of 1.

Layer	Configuration
Conv3D (3 x 3 x 3, 3 $\rightarrow$ 32)	stride = 2, padding = 1
BatchNorm3D	
Conv3D (3 x 3 x 3, 32 $\rightarrow$ 64)	stride = 2, padding = 1
BatchNorm3D	
Conv3D (3 x 3 x 3, 64 $\rightarrow$ 128)	stride = 2, padding = 1
BatchNorm3D	
GlobalAvgPool3D	
Linear	

Table 2. 3D-CNN Model Architecture

The last model we use is *3D Residual Network CNN (R3D-CNN or ResNet3D)* [10]. The architecture of R3D-CNN is based on stacking multiple layers of 3D convolutions interspersed with pooling layers and activation functions. Residual blocks are used throughout the network to maintain performance in deeper networks by adding the input of a layer to the output of a subsequent layer. The ability to process both spatial and temporal dimensions simultaneously makes R3D-CNNs highly effective for video data. In our project, we leverage PyTorch's pretrained 18-layer R3D model,[1] finetuning the final layer to adapt to our highlight detection task.

3.2. Loss Function

We use Binary Cross-Entropy Loss in all models. BCE is a loss function commonly used for binary classification tasks. It measures the performance of a classification model whose output is a probability value between 0 and 1.

$$BCE(y, \hat{y}) = -(ylog(\hat{y}) + (1-y)log(1-\hat{y}))$$

Additionally, due to the imbalanced data issue, we also use a weighted Binary Cross-Entropy loss. WBCE introduces weights to penalize the errors of one class more than the other, thereby giving the minority class more importance during training.

$$WBCE(y, \hat{y}) = -(w_1 ylog(\hat{y}) + w_0(1-y)log(1-\hat{y}))\ ^{2}$$

[1] https://pytorch.org/vision/main/models/generated/torchvision.models.video.r3d_18.html

[2] https://orchardbirds.github.io/bokbokbok/reference/eval_metrics_binary.html

where w_1 is the weight for the positive class, w_0 is the weight for the negative class, y is the true label, and $\hat{y}$ is the predicted probability of the sample being a positive class.

4. Dataset

4.1. Data Source

We downloaded, preprocessed, and labeled 25 full soccer match replays ourselves from YouTube. These game replays are public videos from the official YouTube accounts of the English Football Association Challenge Cup,[3] Italian Supercoppa and Italian Serie A.[4] For labeling, we also relied on the highlight videos uploaded by these official accounts. To enable the preprocessing and labeling steps later, we used the following selection criteria: (a) The game must consistently display timer on the top left corner. (b) The game has an official extended highlight longer than 10 minutes. We used extended highlights rather than short highlights that are generally around 3 minutes so that we could identify a larger set of highlight-worthy clips. (3) The replay video must have a resolution of at least 360p. Events that were being labeled as highlights included goals, shots, fouls, yellow/red cards, free kicks, dangerous passes, counterattacks, impressive personal skills or team tactics, etc.

We randomly assigned each game to the training, validation, and test datasets using a 15-5-5 split ratio.

4.2. Preprocessing

We preprocessed each of the 25 games as follows. First, we manually edited the game replay to split every replay into two 45-minute videos, each corresponds to one half of the soccer match. To streamline labeling, we ignored injury time[5] and extra time after 90 minutes. We then splitted each 45-minute video into 5-second clips that we later would label as highlight-worthy or not. We chose the 5-second length because we have observed the typical clip length in a professional soccer highlight video is 5-15 seconds. We use the lower end to allow for more training examples. This step resulted in 1,080 video clips per game, or 27,000 clips in total across training, validation, and test datasets.

To process the clips into formats compatible with our models, we extracted image frames at 1 frame per second, so each video clip corresponds to 5 image frames. The extraction resulted in 5,400 images per game, or 135,000 images in total across training, validation, and test datasets.

4.3. Labeling

We labeled each clip 1 as highlight-worthy and 0 as not. To do this, we played through the extended highlight video for each game and manually documented all timestamps of the highlight clips. Based on these timestamps, we then mapped the highlights to the clips we have processed and labeled the corresponding clips as highlight-worthy. All 5-second clips that were either fully or partially included in the documented timestamps were considered highlight-worthy. All other clips were assigned with a label of 0. Figures 2 and 3 present examples of frames labeled as class 0 and 1.

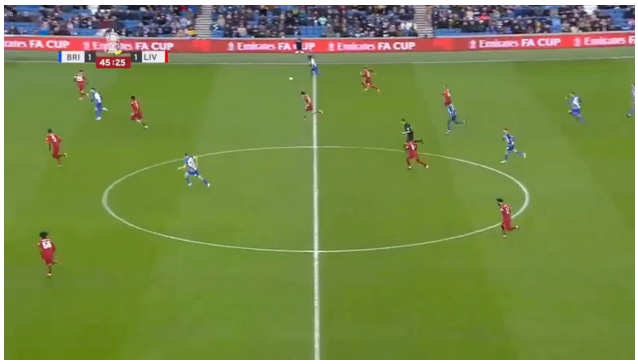

Figure 2. Class 0: A random moment not included in highlight

Figure 3. Class 1: A scoring moment included in highlight

4.4. Augmentation

We apply the following data augmentation steps. First, with a probability of 0.5, we randomly apply color jitter with a brightness of 0.4, contrast of 0.4, saturation of 0.4, and hue of 0.1. Second, we randomly apply grayscale with a probability of 0.2. For each game in the training set, teams have different colors of jerseys. We perform these two steps to improve our models' robustness to color variation such that they would generalize better for unseen data.

To standardize all data, we also resized the image frames. To finetune Vision Transformer, we resized the frames to 224x224. For all other models, we resized the shorter edge to 112. Finally, we normalized the frames by subtracting the mean RGB values and dividing by the standard deviation of each RGB value.

[3] https://www.youtube.com/@thefacup

[4] https://www.youtube.com/@seriea

[5] A short period of time added to the end of each half to compensate for time lost when play was stopped to handle player injury. It typically ranges from 1 to 10 minutes per half.

4.5. Undersampling

After labeling, the ratio between number of clips in class 0 and class 1 is 21:1. The nature of our task, selecting only a small portion of clips to include in the highlights, resulted in an imbalanced dataset. To reduce the bias towards the majority class 0 in this binary classification task, we used random undersampling technique to resample from class 0 such that the ratio between class 0 and 1 becomes 5:1.

5. Experiments and Results

5.1. Hyperparameters

To optimize the performance of our models, we conducted a comprehensive hyperparameter search. Specifically, we explored a wide range of learning rates, including values of 1e-3, 2e-3, 1e-4, 2e-4, 1e-5, 2e-5, and 5e-5. Additionally, we experimented with two popular optimizers: Adam and Stochastic Gradient Descent (SGD). Given the imbalance in our dataset, we addressed this issue by undersampling non-highlight frames, retaining only 5/21 of the original frames. Furthermore, we investigated two loss functions: binary cross-entropy and weighted cross-entropy, the latter assigning higher weights to highlight frames to mitigate class imbalance. Our parameter search revealed that employing the Adam optimizer, utilizing weighted cross-entropy loss, and undersampling non-highlight frames at a rate of 5/21 consistently yielded optimal performance across all models.

Model	Learning Rate	Batch Size
2D-CNN	1e-4	32
3D-CNN	1e-4	32
R3D	2e-5	16
ViT	2e-5	16

Table 3. Best Hyperparameter Combinations

5.2. Evaluation Metrics

We evaluate the performance of our models using several metrics: accuracy, precision, recall, and F1-score. These metrics provide a comprehensive view of model performance. Accuracy measures the proportion of correctly classified clips out of the total number of clips. It is a general measure of how well the model performs overall. Precision measures the proportion of true positive predictions out of all positive predictions made by the model. It indicates how many of the clips identified as highlights are actually highlights. Recall measures the proportion of true positive predictions out of all actual positive instances. F1-Score is the harmonic mean of precision and recall. It is especially useful when the class distribution is imbalanced, as it provides a more comprehensive evaluation of the model's performance.

In particular, we note that recall is important in our analysis. Recall measures the ability of the model to find all relevant instances in the dataset, i.e., the true positive rate. For highlight detection, this means capturing all the significant moments in the game. Missing key highlights would be a significant drawback since the primary goal is to ensure that all exciting and important moments are captured. Finally, we conduct a manual review of the generated highlights for qualitative evaluation, assessing the relevance and coherence of the detected events.

The equations for the quantitative metrics are as follows:

- *Accuracy* = $\frac{TP+TN}{TP+TN+FP+FN}$
- *Precision* = $\frac{TP}{TP+FP}$
- *Recall* = $\frac{TP}{TP+FN}$
- *F1-Score* = $\frac{2*Precision*Recall}{Precision+Recall}$

5.3. Results

Metrics	2D-CNN	ViT	3D-CNN	R3D
Accuracy	**0.740**	0.664	0.552	0.635
Recall	0.149	0.272	**0.555**	0.506
Precision	0.174	0.174	0.198	**0.230**
F1-score	0.161	0.212	0.292	**0.312**

Table 4. Evaluation Results

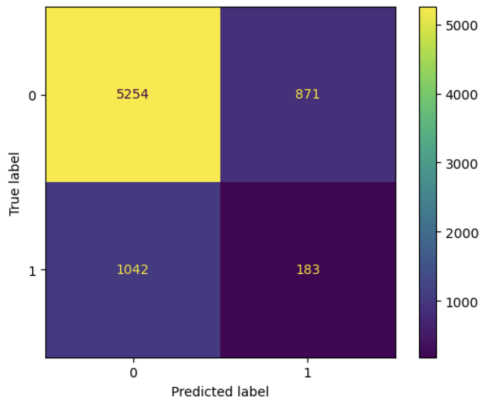

Figure 4. Confusion matrix for 2D-CNN

Table 4 presents the evaluation results. Figures 4 to 7 display the confusion matrices for all four models. We note that our model performances are lower than those in our milestone report. We investigated this issue and conclude that our models significantly overfitted the smaller training dataset we relied upon in the milestone. Using the full dataset, we report the following results.

Result 1: Both 3D models outperform 2D models in terms of recall, precision, and f1-score. We expect this result because 3D models are more capable of capturing the

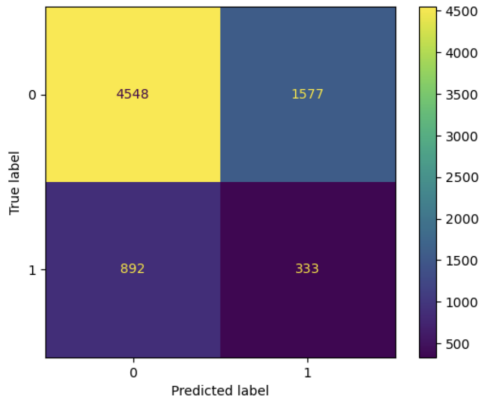

Figure 5. Confusion matrix for ViT

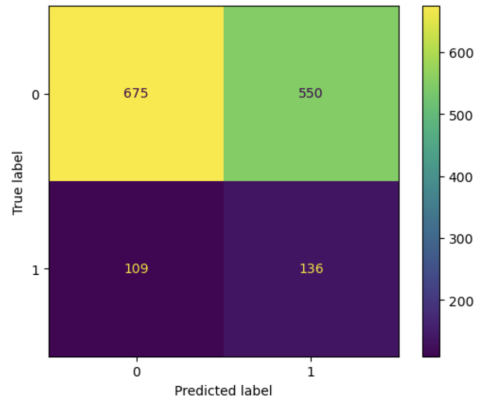

Figure 6. Confusion matrix for 3D-CNN

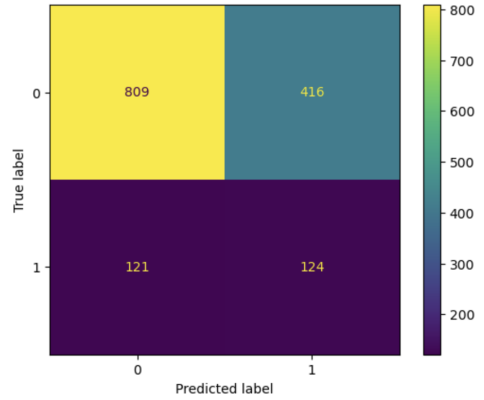

Figure 7. Confusion matrix for R3D

dynamics in the clips. Instead of analyzing features at the frame level, 3D models could also identify features at the five-frame continuous clip level. Thus, they are better suited to detect whether the live action is fast-pacing enough to be included in the highlight. This task is challenging for 2D models, even using a finetuned state-of-the-art Vision Transformer model. It is reasonable because even for human evaluation, it would be difficult to determine whether a particular frame should be included in the highlight or not without seeing the relevant clip.

Result 2: Both pretrained models outperform models we trained from scratch in terms of f1-score, but not by much. We expect the finetuned models to perform better because they are larger models that have been previously trained on more data. However, it was surprising to see that the improvements were marginal. This result could be due to the small size of our dataset, especially the relatively small set of class 1.

Result 3: Our best models are decent at identifying highlights (higher recall) but poor at filtering out non-highlights (lower precision). While 3D-CNN and R3D achieved greater than 0.5 recall, their precision was only around 0.2. This result implies that our model could label at least half of the highlight-worthy clips correctly, but it generally would also include many non-highlight-worthy clips in the highlight pool. Upon qualitative evaluation, we found this result explainable for our task. Solely relying on video clips without commentary or in-game statistics, it is often inherently ambiguous to classify a clip. For example, Figure 8 belongs to a clip whose true label is 0 but predicted label is 1 by R3D. A player in red is dangerously in the box and could have scored a goal here. This frame appears quite similar to Figure 3 and without any context, it is not unreasonable to label it as highlight-worthy. To address this issue, future work could explore multi-modal models to incorporate additional information such as commentary transcription to better detect events and live stadium sound to better capture fan reactions.

Figure 8. A frame falsely classified as highlight-worthy

On the other hand, we also qualitatively evaluated cases where a clip was misclassified as 0. Figure 9 provides an example. This frame belongs to a clip that marked the start of a counter attack which led to a goal. This clip is included in the official match highlight, but the exciting action of the goal does not happen until 10 seconds later. While the following two clips were indeed classified as 1 by R3D, it is difficult for the model to recognize that this clip could be included too. Future work could conduct additional tests

using different clip lengths such as 10 or 15 seconds. Given that the clip splitting and frame extraction processes alone took approximately 10 hours for each clip length, we did not conduct the experiment for this study.

Figure 9. A frame falsely classified as not highlight-worthy

Given these qualitative evaluations and our emphasis on recall more than precision, our models decently achieved the goal of our task.

6. Conclusion and Future Work

Throughout this project, we collected our own dataset by preprocessing 25 soccer match videos. We splitted each soccer frame into small clips of 5 seconds each, where each splitting of a 45-minute video took 10 minutes to complete. Subsequently, we splitted each clip into 5 frames, and employed CNN and Transformer models to classify these frames as containing a highlight or not. To achieve better results, we performed data augmentation, data undersampling, and hyperparameter tuning. We found that the 3D-CNN-based models outperformed the 2D-CNN-based models, and achieved a recall of over 0.5. However, despite our efforts, it was difficult to get any model to achieve the desired level of performance in terms of both a high recall and a high precision. The inherent complexity of soccer gameplay, with its myriad of subtle nuances and dynamic interactions, posed significant obstacles to our classification efforts.

There are several things we would like to explore for future work. First, we observed that we performed careful data labeling and extensive fine-tuning of the different models, the results were still not satisfactory in terms of capturing highlights from a game. We believe this is due to the inherent nature of the data itself. For example, some actions or events such as counterattacks or dangerous passes are included in official highlights, but it is extremely difficult for the models to discern these as important events. On the other hand, the model may classify events such as many players being close to each other as an important event, while in reality this should not be classified as a highlight.

Hence, one thing we would like to explore is the use of multi-modals to achieve our task. In our project, mainly due to time limits, we only used computer vision models that were designed for classification tasks. For future work, given a much longer timeframe and more compute resources, we could use the reporter commentaries or match summaries from the videos, feed these text into a language model, and therefore achieve a better result by selecting more accurate frames. Audio data that capture live fan reactions could also provide useful features for our tasks. Furthermore, with additional time we envision a more comprehensive exploration of model architectures and training methodologies. Experimenting with state-of-the-art techniques, such as self-supervised learning or attention mechanisms tailored specifically for sports video analysis, could potentially unlock new avenues for improving highlight detection accuracy.

7. Contributions and Acknowledgements

The two team members contributed equally to this project. At the data preparation stage, Mike Yang worked on data selection and collection, and Fang Shu worked on preprocessing and labeling. At the experiment stage, Mike mainly focused on training while Fang mainly focused on inference and evaluation. Both worked closely with each other to draft the final report.

We would like to thank Nikil Ravi, our project mentor, for his generous assistance during office hours and the invaluable feedback provided on both our proposal and milestone submissions.

References

[1] N. Darapaneni, P. Kumar, N. Malhotra, V. Sundaramurthy, A. Thakur, S. Chauhan, K. C. Thangeda, and A. R. Paduri. Detecting key soccer match events to create highlights using computer vision, 2022.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[3] B. Fakhar, H. Rashidy Kanan, and A. Behrad. Event detection in soccer videos using unsupervised learning of spatio-temporal features based on pooled spatial pyramid model. *Multimedia Tools and Applications*, 78(12):16995–17025, 2019.

[4] M. Godi, P. Rota, and F. Setti. Indirect match highlights detection with deep convolutional neural networks. In *New Trends in Image Analysis and Processing–ICIAP 2017: ICIAP International Workshops, WBICV, SSPandBE, 3AS, RGBD, NIVAR, IWBAAS, and MADiMa 2017, Catania, Italy, September 11-15, 2017, Revised Selected Papers 19*, pages 87–96. Springer, 2017.

[5] S.-K. Kang and J.-H. Lee. An e-sports video highlight generator using win-loss probability model. pages 915–922, 03 2020.

[6] M. H. Kolekar and S. Sengupta. Bayesian network-based customized highlight generation for broadcast soccer videos. *IEEE Transactions on Broadcasting*, 61(2):195–209, 2015.

[7] D. Z. Liu, A. Bratko, J. Prevc, L. Pataky, M. Noori, S. Sathyanarayana, and U. Lipovsek. Predicting soccer goals in near real time using computer vision, 2020.

[8] O. A. Nergård Rongved, M. Stige, S. A. Hicks, V. L. Thambawita, C. Midoglu, E. Zouganeli, D. Johansen, M. A. Riegler, and P. Halvorsen. Automated event detection and classification in soccer: The potential of using multiple modalities. *Machine Learning and Knowledge Extraction*, 3(4):1030–1054, 2021.

[9] J. Park, Y. Jwa, J. Kwak, J. Lim, and S. Kim. Automatic highlight generation of soccer videos. In *2023 14th International Conference on Information and Communication Technology Convergence (ICTC)*, pages 1867–1871. IEEE, 2023.

[10] Z. Qiu, T. Yao, and T. Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017.

[11] K. Tang, Y. Bao, Z. Zhao, L. Zhu, Y. Lin, and Y. Peng. Autohighlight : Automatic highlights detection and segmentation in soccer matches. pages 4619–4624, 12 2018.

[12] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, Z. Yan, M. Tomizuka, J. Gonzalez, K. Keutzer, and P. Vajda. Visual transformers: Token-based image representation and processing for computer vision, 2020.

[13] W. Zhao, Y. Lu, H. Jiang, and W. Huang. Event detection in soccer videos using shot focus identification. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 341–345. IEEE, 2015.