Soccer Event Detection

Jad YEHYA

Abstract

Soccer event detection is a challenging task in sports analysis, which involves classifying different events occurring during soccer matches. In this paper, we explore the application of deep learning methods for soccer event detection using a dataset of soccer images. We investigate the performance of four deep learning models: ViT, BCNN, NFnet, and VQ-VAE, in classifying soccer events within images. The dataset we use contains annotated images representing various soccer events, providing valuable training and evaluation material. We conduct experiments and evaluate the models based on standard performance metrics. Our results demonstrate the potential of deep learning models for soccer event detection, showcasing their effectiveness and limitations. The findings of this study contribute to the understanding of deep learning techniques in automating event classification in soccer imagery.

1 Introduction

Detecting events in soccer videos or images poses a significant challenge due to the diverse nature of events and the complex visual characteristics of the game. Various approaches have been proposed to address this task, with each method employing different strategies to detect specific events. This work builds upon the research conducted by [1], which has made notable advancements in soccer event detection. Additionally, we utilize the SEV dataset 3, which provides annotated images capturing a wide range of soccer events, enabling effective training and evaluation of our models. We chose to keep the same general architecture because the paper already solves a lot of problems past researches suffered from. This is why we haven't reinvented any new architecture.

While our results may not have achieved the same level of performance as the previous work, we have made important modifications that have the potential to improve the results. Notably, we refrain from employing data augmentation and regularization techniques for the pre-trained models. This decision is based on the findings presented in [2], which indicate that such techniques may weaken the results for vision pre-trained models. By avoiding these techniques, we aim to maintain the integrity and effectiveness of the models in our soccer event detection task.In the following sections, we will discuss the dataset used, the deep learning models employed, and the experimental setup conducted to evaluate their performance.

2 Methodology

The paper [1] already aims to solve the main problems of other approaches so I took the same general architecture as them which is shown in Figure 1 but added a module (3). The idea is to divide the problem into 4 separate tasks : (1) Detecting football from other images, (2) Detecting football events from football images, (3) Using a CNN classifier to detect the type of event, (4) Using a fine-grain classifier to differentiate between yellow and red card. We chose to add a module so we decompose a task into a binary classification task.

The pipeline is straightforward, if an image is considered football by the VQ-VAE, then it is passed to the next model which checks if it is an event or not. If it is, we give it to the next model which checks the type of event. If it is a card, we give it to the last model which checks if it is a yellow or red card.

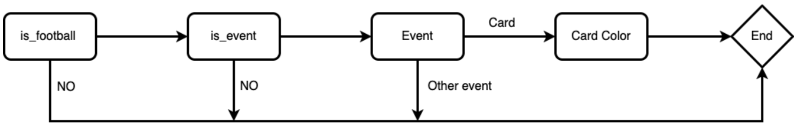

Fig. 1 Architecture

We added the binary module of event / not event in the process to divide the general problem and make a special module for it.

2.1 Detecting football from other images

The first task is to detect football from other images. This is done by using a vector-quantized variational autoencoder (VQ-VAE) [3] [4] and measuring the reconstruction error. The VQ-VAE is only trained on football images so naturally, it will be able to reconstruct football images better than other images, therefore, the reconstruction error will be lower for football images than other images. The VQ-VAE is trained to minimize the reconstruction error. The reconstruction error is calculated by using the mean squared error (MSE) between the input image and the reconstructed image. We used here a VQ-VAE instead of a VAE because it has been shown that it solves the problem of 'posterior collapse' and it also has better results than a VAE. For this model, we used a hidden layer and a latent space size of 64. This allows us to compress the data but to still have enough information to reconstruct the image. We also used a codebook size of 512. The codebook is the number of vectors that the input is quantized to. The codebook size is chosen to be 512 because it is the smallest codebook size that gives good results. We did other experiments with other sizes for the hidden layer and the latent space dimension but it didn't give better results.

2.2 Detecting football events from football images

The second task is to detect football events from non football events. For this task, we use a pre-trained model that we fine-tune. The pre-trained model is NFNet-F0 [5] which is trained on ImageNet [6]. We fine-tune it by using the methods described in [2] and in section 4. The model is trained to minimize the cross-entropy loss.

2.3 Using a ViT classifier to detect the type of event

For this we use a Visual Transformer (ViT) [7] [8] to extract features from the images. The ViT is pre-trained on ImageNet-21k [6] and then fine-tuned on the dataset we have. The ViT is trained to classify the images into 6 classes : (0) Cards, (1) Corner, (2) Free-Kick, (3) Penalty, (4) Tackle, (5) To_Subtitue. It is trained to minimize the cross-entropy loss. The Transformer used is the google/vit-base-patch16-224-21k [7]. We chose this model because it is the smallest ViT model from Google on HuggingFace and it is also has really good results. We started with the base model and then used the raw output of the model and the `cls token` to the classification head which consists of a `Linear` layer that outputs 256 features to a `ReLU` activation function and then another `Linear` layer that outputs 6 features corresponding to the classes.

2.4 Using a fine-grain classifier to differentiate between yellow and red card

For the last part of the pipeline, we use a fine-grain classifier to differentiate between yellow and red card. We used a bilinear CNN (BCNN) [9] to extract features from the images and we then use the classification head we implemented to classify the images into 2 classes : (0) Yellow, (1) Red. It is again trained to minimize the cross-entropy loss. This part is the most challenging and is the weak link of the model because the difference between a yellow and red card is very subtle and the images are very similar. This can be seen when we look at the accuracy which is only 73.05% on the validation set.

3 Dataset

The dataset used is the SEV dataset [1] which contains 11 × 5500 images of football of every event type. Also, we have 2 smaller datasets called `Soccer` and `Event` that contain 1200 images each of Soccer images without event and Soccer images of Events. We divided the dataset with a classical train/validation split of 80% for the train set and 20% for the validation set. We used the validation to validate the model during training. The dataset is very small and we can't use data augmentation and regularization because it weakens the results as seen in [2].

4 Experiments and Results

In general, the training method is similar for every model, we used the following parameters. If any change was made, it will be specified in the corresponding section.

Table 1 Hyperparameters used

Batch Size	Optimizer	Learning Rate	Loss Criterion	Transforms
64[1]	Adam	1e-3	Cross entropy	Resize to 224*224, Normalize

[1] Tests with 128 were made when it was possible for the GPU.

We also used early stopping when the test accuracy wasn't good for more than 4 epochs.

4.1 Detecting football from other images

For the VQ-VAE we don't really have an accuracy but we have reconstruction error that we are aiming to minimize. We saw that the average reconstruction loss was 0.16 for football images, we allow ourselves 5% of error so we set the threshold to 0.17. If the reconstruction error is lower than the threshold, we classify the image as a football image, otherwise, we classify it as a non-football image.

In figure 2 we can compare the ground truth image and the reconstructed image. The reconstructed image is similar to the ground truth image but it is not perfect. The reconstructed is a little bit blurry but the colors are reproduced well and we can still see the main features of the image and understand that it is football. We also downsampled the images by a factor of 2 and ended up with images of size $(112 * 112 * 3)$ before feeding it to the model, this does almost not affect the performance of the model but sped up the training time.

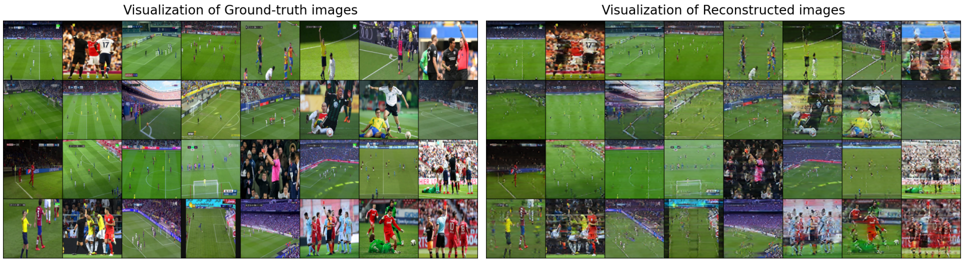
Visualization of Ground-truth images
Visualization of Reconstructed images

Fig. 2 Ground truth image (left) and reconstructed image (right)

Also, we can see in figure 3 the evolution of the different losses.

For this model, we used a batch size of 128 and the mean squared error (MSE) as our optimization criterion. It was trained on 10 epochs and it took around 3 hours to train.

4.2 Detecting football events from football images

As stated in 2.2, we fine-tuned a NFNet-F0. Using the same hyper-parameters as in table 1. We obtained really good results with an accuracy of 99.03%. We can see in figure 4 the evolution of the loss during training.

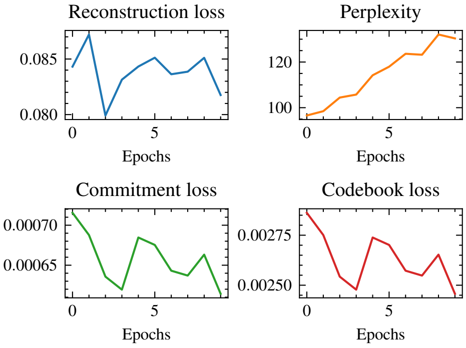

Fig. 3 Evolution of the different losses

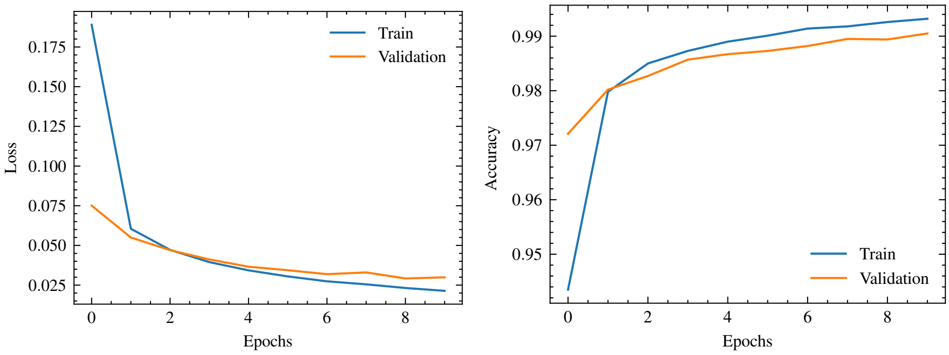

Fig. 4 Evolution of the loss and accuracy during training of the NFNet

4.3 Detecting the type of event

For this task, we used a ViT-B/16 model. We used the same hyperparameters as in table 1 but used the preprocessor made by google instead of our transforms. We obtained an top-1 accuracy of 93.5% and an F1 score of 84.3%. We can see in figure 5 the evolution of the loss during training. We also used gradient clipping with a maximum norm of 1.0 to avoid exploding gradients.

4.4 Detecting the type of card

This last task was the key-point of the improvements in the paper [1] so we naturally kept it. We used the same hyperparameters as in table 1. The results were not as good as the ones in the paper because of training time and model size but also because this fine-grain task is really hard. We obtained an accuracy of 73%. We can see in figure 6 the evolution of the loss during training.

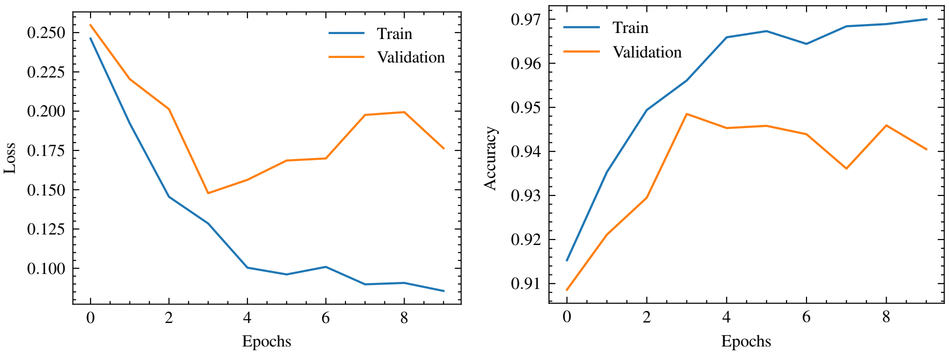

Fig. 5 Evolution of the loss and accuracy during training of the visual transformer

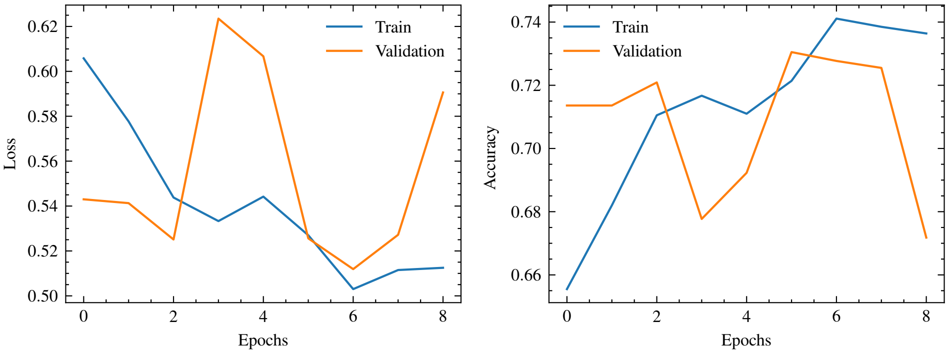

Fig. 6 Evolution of the loss and accuracy during training

4.5 Final model results

The final model is just an assembly of the previous components presented in section 2. It allows us to classify in 9 classes : (1) Not Event, (2) Card, (3) Corner, (4) Tackle, (5) Free Kick, (6) Penalty, (7) Yellow Card, (8) Red Card, (9) To Substitue. The accuracy at random would be $1/9 = 0.11$. We can see in table 2 the results of the final model. We don't obtain good results generally even though every component is working well. The biggest mis-classification is the model classifying 'Card' events in 'To_Subtitue'. Both images showcase a staff member holding an object up. For example, the ViT module on its own has a really good accuracy but the final pipeline the results are not good. This could be because every module has some error and it propagates through the pipeline.

Table 2 Final model results

Task	**Top-1 Accuracy**	**Precision**	**Recall**	**F1**
Detecting the type of event (ViT)	0.935	0.879	0.878	0.84
Final model	0.28	0.27	0.281	0.24

5 Conclusion

In this study, we explored the application of deep learning models for soccer event detection using a dataset of annotated soccer images. The four models, ViT, BCNN, NFnet, and VQ-VAE, were investigated for their performance in classifying soccer events within images. Through a series of experiments and evaluation, we gained insights into the capabilities and limitations of these models in automating event classification in soccer imagery. The results of our experiments indicate that deep learning models can effectively classify soccer events with a reasonable level of accuracy.In conclusion, this project contributes to understand deep learning techniques for soccer event detection. The findings highlight the effectiveness of deep learning models, particularly ViT, BCNN, NFnet, and VQ-VAE, in automating event classification in soccer imagery.

6 Future Work

Multiple improvements can be made for us to get better results. First, we should've kept a final test set to test that the model has never seen, In our case, we tested the models but the train/test split was random and not saved. Also, we could've saved more metrics while training the models to better understand what is happening, for example, to compute the precision, recall, and F1 score, confusion matrix, etc. for every model and not the full task. The main reason we didn't is because training time is very long, it would've required to start training again from scratch. We could also use a better pre-trained model for the second task, for example, `NFNet-F4` or `NFNet-F5`.

References

[1] Karimi, A., Toosi, R., Akhaee, M.A.: Soccer Event Detection Using Deep Learning. arXiv. arXiv:2102.04331 [cs] (2021). https://doi.org/10.48550/arXiv.2102.04331 . http://arxiv.org/abs/2102.04331 Accessed 2023-06-09

[2] Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., Beyer, L.: How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers. arXiv. arXiv:2106.10270 [cs] (2022). https://doi.org/10.48550/arXiv.2106.10270 . http://arxiv.org/abs/2106.10270 Accessed 2023-05-18

[3] Oord, A.v.d., Vinyals, O., Kavukcuoglu, K.: Neural Discrete Representation Learning. arXiv. arXiv:1711.00937 [cs] (2018). http://arxiv.org/abs/1711.00937 Accessed 2023-06-05

[4] Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. arXiv. arXiv:1312.6114 [cs, stat] version: 10 (2022). https://doi.org/10.48550/arXiv.1312.6114 . http://arxiv.org/abs/1312.6114 Accessed 2023-05-24

[5] Brock, A., De, S., Smith, S.L., Simonyan, K.: High-Performance Large-Scale Image Recognition Without Normalization. arXiv. arXiv:2102.06171 [cs, stat]

(2021). https://doi.org/10.48550/arXiv.2102.06171 . http://arxiv.org/abs/2102.06171 Accessed 2023-06-06

[6] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009). Ieee

[7] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv. arXiv:2010.11929 [cs] version: 2 (2021). http://arxiv.org/abs/2010.11929 Accessed 2023-05-18

[8] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need. arXiv. arXiv:1706.03762 [cs] (2017). https://doi.org/10.48550/arXiv.1706.03762 . http://arxiv.org/abs/1706.03762 Accessed 2023-06-09

[9] Lin, T.-Y., RoyChowdhury, A., Maji, S.: Bilinear CNNs for Fine-grained Visual Recognition. arXiv. arXiv:1504.07889 [cs] version: 6 (2017). http://arxiv.org/abs/1504.07889 Accessed 2023-05-30