17th International Conference on Machine Vision Applications (MVA)
Fully Online, July 25–27, 2021.

Action Spotting and Temporal Attention Analysis in Soccer Videos

Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi
Chubu University
{himi1208, hirakawa@mprg.cs, takayoshi, fujiyoshi@isc}.chubu.ac.jp

Mitsuru Nakazawa, Yeongnam Chae, Björn Stenger
Rakuten Institute of Technology, Rakuten Group, Inc.
{mitsuru.nakazawa, yeongnam.chae, bjorn.stenger}@rakuten.com

Abstract

Action spotting is the task of finding a specific action in a video. In this paper, we consider the task of spotting actions in soccer videos, e.g., goals, player substitutions, and card scenes, which are temporally sparse within a complete game. We spot actions using a Transformer model, which allows capturing important features before and after action scenes. Moreover, we analyze which time instances the model focuses on when predicting an action by observing the internal weights of the transformer. Quantitative results on the public SoccerNet dataset show that the proposed method achieves an mAP of 81.6%, a significant improvement over previous methods. In addition, by analyzing the attention weights, we discover that the model focuses on different temporal neighborhoods for different actions.

1 Introduction

Action spotting is the task of finding a specific action in a video and is an important part of video understanding. With the recent explosive growth in the number of online videos, efficient methods for search, classification, and further analysis are needed.

A number of video datasets have been introduced, where each video contains multiple different actions [1, 2, 3, 4, 5, 6, 7, 8]. Each video is annotated with action labels and the corresponding time instances or intervals. One challenge is that in many cases actions are temporally sparse in a video, such as goals, substitutions and card scenes in SoccerNet [7]. Much of the video contains game play without any of these actions of interest: The average number of goals in professional soccer games is 2-3 (2.6 at the most recent World Cup [9]). In order to detect such sparse events, it is important to understand the temporal context of the action. For example, a goal can be recognized after the shot itself, when players celebrate and a slow motion replay is shown. In a substitution scene, players leave and enter the pitch shortly after the referee raises the board indicating the player numbers.

Prior work proposed efficient action spotting methods for soccer video [7, 10, 11], which addressed the data imbalance and sparsity of action classes by using appropriate loss functions and temporal pooling of video features extracted by convolutions. For sparse action spotting it is important to capture different related scenes, *i.e.*, the temporal context of each action, which has not fully been captured in prior work.

The goal of this paper is to improve the accuracy of action spotting by allowing the model to automatically use the correct temporal features. Typically, temporal features are extracted using an LSTM auto-encoder or CNN convolutions in the temporal direction. However, LSTM auto-encoders are known to struggle with capturing features over longer time periods [12, 13]. CNN convolutions along the time axis can be considered temporal pooling as in previous work. In this work, we adapt the transformer model [14] to model context over longer time periods in SoccerNet. The transformer captures features by calculating the similarity of feature vectors at different times. Moreover, it is possible to analyze the temporal attention for action spotting by analyzing the attention weight obtained by the transformer. This allows us to clarify the explanation for the action predicted by the model.

In summary, the main contributions of this work are:

- By using temporal attention of each action our transformer model improves the action spotting accuracy using a transformer, achieving 81.6% mAP, a significant improvement over prior work, shown in Section 4.2.
- To the best of our knowledge, this work analyzes temporal attention for action spotting on soccer video for the first time, see Section 3.2.
- As a result of the internal analysis of the transformer, we discovered that the temporal attention was different for each action label, shown in Section 4.2.

2 Related Work

In this section, we review recent work on video understanding with a focus on action spotting in sports videos.

2.1 Video Understanding

Video understanding includes a number of approaches, such as action recognition [1, 15, 16], spatio-temporal lo-

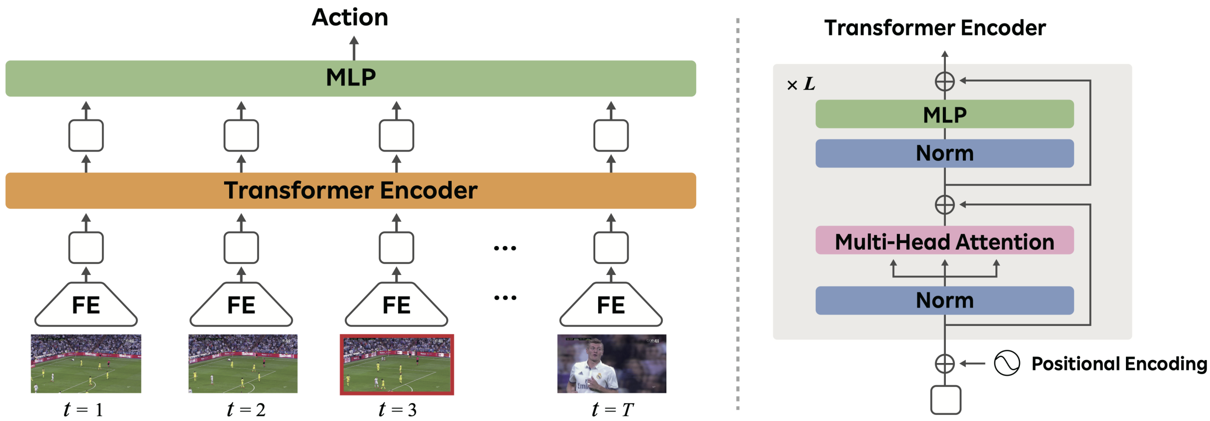

Figure 1: **Proposed Network Architecture**. Our model adopts a transformer as a backbone model. The red rectangle frame shows the annotated frame for an action label. The transformer spots appropriate labels in soccer videos using a ResNet-based features extractor (FE), in our case applied to 120 frames extracted every 0.5 seconds. The images are cited from [7].

calization [17, 18, 19, 20], and video classification [6, 17, 21, 22]. Video classification is the task of predicting a label that is relevant to the video. Most recent work on video classification is based on 3D convolutional networks and recurrent neural networks. Donahue *et al.* [22] proposed a method of propagating CNN feature vectors in the temporal direction using an LSTM. Karpathy *et al.* [6] used a CNN architecture to learn spatio-temporal features for action classification. Much of the video classification work assigns a single label to the entire video.

Recently, action spotting has been studied as the task of predicting labels of a specific scene in the video [7, 8, 10, 11, 23]. Action spotting is important for long-term summarization, *e.g.*, of cooking or sports videos. Giancola *et al.* [7] provide a baseline of action spotting tasks on soccer video by introducing pooling and context gating layers [24, 25, 26, 27]. Tomei *et al.* [11] proposed a method for action spotting and approximate temporal offset regression by taking the maximum features in the time direction after convolution. These methods improve the accuracy by capturing temporal features, however, they do not consider different related scenes for each action in the input video. Here we consider action-related scenes by using a transformer to calculate the similarity between different scenes for each action.

2.2 Datasets

Many sports video datasets have been published for action detection and classification [3, 4, 6, 8, 28] as well as action spotting [7, 28, 29]. The NCAA dataset [8] contains basketball footage and is annotated with 11 action classes. The Sports-1M dataset [6] contains 487 classes of sports videos. Datasets for action spotting are annotated with momentary action in a video. SoccerNet [7] contains soccer videos labeled with 4 actions. SoccerDB [28] additionally contains annotations for behavior recognition and object detection tasks.

3 Method

In this section, we describe the model for sparsely annotated data and the temporal attention analysis.

3.1 Overview

The architecture of the proposed network is illustrated in Figure 1. Our method consists of a feature extractor and a transformer encoder. The transformer receives as input a sequence of embedding vectors. First, we convert 2D images $\mathbf{x}_t \in \mathbb{R}^{H\times W\times C}, t = 1, 2, ..., T$ into d-dimensional feature vectors $\mathbf{h}_t \in \mathbb{R}^{1\times d}, t = 1, 2, ..., T$ using the feature extractor $f(\cdot)$:

$$\mathbf{h}_t = f(\mathbf{x}_t). \tag{1}$$

$(H \times W \times C)$ is the resolution of the original image, and T are the input time steps. As proposed in [14] we inject temporal information by adding positional encoding vectors to each feature vector. These vectors are of the same dimension, d_{model}, as the feature vectors, and the components are sinusoids of different wavelength:

$$PE_{(t,2i)} = \sin\left(\frac{t}{10000^{\frac{2i}{d_{\text{model}}}}}\right), \tag{2}$$

$$PE_{(t,2i+1)} = \cos\left(\frac{t}{10000^{\frac{2i}{d_{\text{model}}}}}\right), \tag{3}$$

where t is the time index and $i \in \{0, ..., \lfloor\frac{d_{\text{model}}-1}{2}\rfloor\}$ is the index of the positional encoding. This results in a sequence, $\mathbf{p}_t$, of feature vectors with embedded positional encodings. The transformer learns latent features for the action labels for each time step via a self-attention mechanism with L layers and N heads. The self-attention first learns the query matrix $Q = \phi_q(\{\mathbf{p}_t\}_{t=1}^T)$, the key matrix $K = \phi_k(\{\mathbf{p}_t\}_{t=1}^T)$ and the value matrix $V = \phi_v(\{\mathbf{p}_t\}_{t=1}^T)$, where ϕ_q, ϕ_k, and ϕ_v are MLP layers. It computes the attention by

$$\begin{aligned} \text{MultiHead}(Q, K, V) &= \phi_o([\text{head}_n]_{n=1}^N), \text{where} \quad (4) \\ \text{head}_n &= \text{Attention}_n(Q, K, V), \quad (5) \end{aligned}$$

ϕ_o is an MLP layer, and the Attention function is the scaled dot-product attention in [14]. The feature vector, obtained by the transformer in each time step, is averaged and passed through an MLP layer $\phi_p(\cdot)$ to obtain the action probability, $\mathbf{y}$, as

$$\mathbf{m} = \frac{1}{T} \sum_t^T \text{MultiHead}(Q_t, K_t, V_t), \quad (6)$$

$$\mathbf{y} = \phi_p(\mathbf{m}). \quad (7)$$

3.2 Temporal attention analysis

We use the attention weights to indirectly obtain explanations for the action predictions. Inspired by ViT [30], which focuses on spatial image features only, we analyze differences between the temporal attention and the frame annotations using attention rollout [31]. Attention rollout is an intuitive way to approximate features of interest using the attention weight from the first layer to the last layer of self-attention. Given a transformer with L layers and N heads, we compute the attention weight $\alpha_{i,j}^{l,n}$ of all layers l, heads n, and time steps i, j. Note that the attention graph for $n \times T^2$ is constructed in each layer l. To compute the attention score A_i for time step i, we first aggregate the attention weight of each head n and add all values of the j-th frame focused on the i-th frame from the attention weight. Finally, the value is calculated as the product over all layers. We select the frame index $\hat{i}$ with the strongest temporal attention by taking the max value over time steps i:

$$A_i = \prod_l \left(\sum_j \sum_n \alpha_{i,j}^{l,n} \right), \quad (8)$$

$$\hat{i} = \arg\max_i (A_i). \quad (9)$$

3.3 Implementation Details

The image dimensions $(H \times W \times C)$ are set to $224 \times 224 \times 3$ by resizing and cropping the input images. We extract features using a ResNet-152 [32], pre-trained on ImageNet [33]. The dimension of feature vectors is reduced to $d = 512$ using Principal Component Analysis. Features are extracted every 0.5 seconds over $T = 120$ time steps, *i.e.*, the network takes as input features computed over 60-second intervals. Our transformer model has 8 heads and 6 layers. The feature vector dimension for self-attention is set to 512. The dimension of the output at each time step in the transformer is set to 4 via a single MLP layer ϕ_p [512×4].

We trained using a cross-entropy loss, and an Adam optimizer [34] with an initial learning rate of 10^{-5}. The model was trained for 400 epochs with a batch size of 16. We handled sparse data by weighting the cross-entropy loss as in [7], which we set to [background, card, substitution, goal] to [0.1, 0.5, 1.0, 1.0].

4 Experiments

In this section, we compare the proposed model with several methods on the SoccerNet dataset.

4.1 Evaluation Protocols

SoccerNet [7] is a dataset annotated with 4 actions (background, card, substitution, and goal) in 500 soccer videos. The background class accounts for about 70% of all data samples. Following [7], we use a 300/100/100 split for training, validation, and testing, respectively. We compare different feature representations and various pooling methods. Mean and max-pooling are pooling methods to obtain a d-dimensional feature vector at each time step. *CNN* denotes a method convolving $d \times T$ dimension in the time direction. *FC* is a method to reshape the $d \times T$ dimension and output the features. *LSTM* is a method to spot via 3 layers (input-LSTM-output) to feature vector taken by a feature extractor. *SoftDBOW* [26], *NetFV* [25], *NetRVLAD* [24], and *NetVLAD* [35] leverage context gating layers, to reweight both the features and the output labels. We use publicly available code to compare these methods with the proposed one. We use classification mAP and AP as evaluation metrics, and analyze the temporal attention for each correctly spotted action by ranking via attention rollout.

4.2 Results

Table 1 shows the mAP and AP scores on SoccerNet. Our method achieves an mAP of 81.6%, an absolute mAP improvement of 26.6% over the next best method in the comparison. The method also achieves the best average precision (AP) for each of the classes. Note that card scenes are detected with lower accuracy than goals or substitutions. We believe this is due to cases when no close-up of the referee is shown and the region where the action takes place is small within the images.

We denote the attention weight obtained by attention rollout the 'attention score'. Figure 2 shows the visualization of the attention score in a transformer. We show that actions were spotted while focusing on different time steps for different action labels. In Fig. 2(a) and (d), we show that the largest attention score is at the time when the referee raises the yellow card. As a result, the annotated label frame and attention match. Fig. 2(b) and (e) show that for substitutions attention is focused on frames before the annotated frame. The model focuses on the board indicating player numbers prior to the annotated frame. Finally, Fig. 2(c) and (f) show that for goals the attention is focused on the frames after the annotated frame, highlighting the replay and player actions after the goal itself.

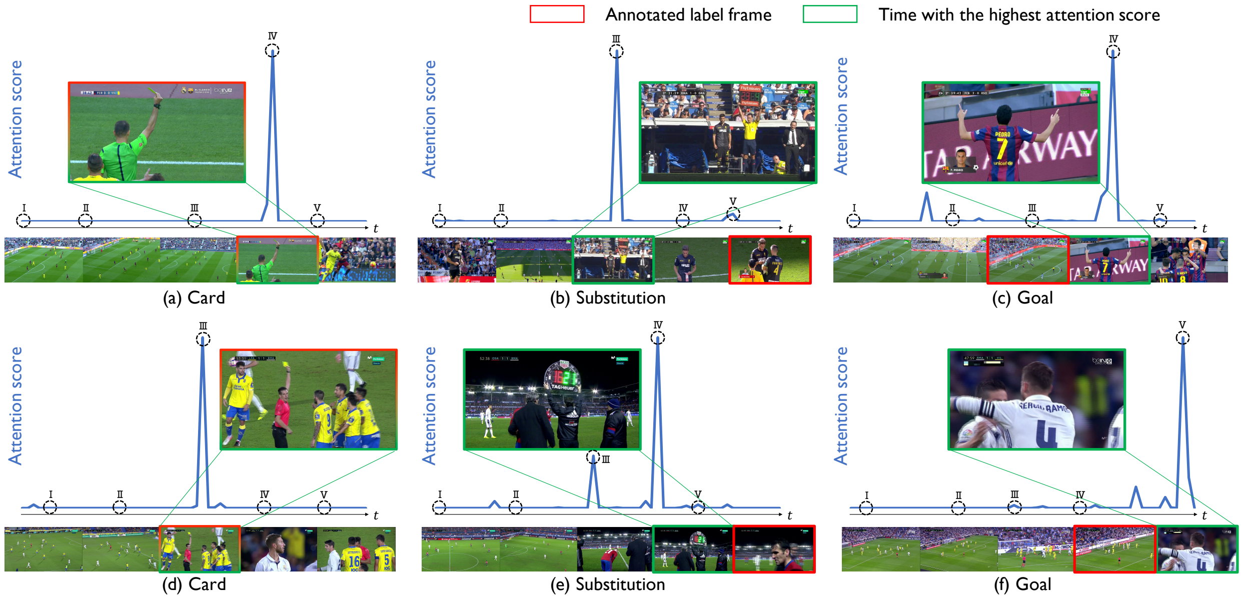

Figure 2: **Visual examples of attention score**. The attention score plotted over time for different scenes and action labels. Red boxes show the annotated frame, green boxes show the time with the highest attention score. Circles (labeled I, ..., V) on the time axis correspond to the five frames below, in left-to-right order. In all cases, the attention score has strong peaks, highlighting discriminative frames for each action. In cases (a) and (d), the two are the same, in the other cases, the highest attention score is at a different time from the annotated frame. The images are cited from [7].

Table 1: **Spotting results (mAP and AP) on SoccerNet**.

Method	mAP	AP		
		Card	Sub	Goal
Mean Pool.	35.1	25.7	38.6	41.1
Max Pool.	52.4	52.4	52.9	51.9
CNN	25.4	21.7	26.6	27.9
FC	52.4	52.4	52.9	51.9
LSTM	48.7	49.9	50.5	45.6
SoftDBOW [26]	48.0	36.0	56.8	51.3
NetFV [25]	52.7	35.0	64.2	58.9
NetRVLAD [24]	52.3	40.5	51.4	55.1
NetVLAD [35]	55.0	44.5	62.6	58.0
Ours	**81.6**	**63.3**	**94.3**	**87.1**

Table 2: **Attention rollout results**. Shown is the average time difference (seconds) between the annotated time and the n-th largest attention weight for each label over all samples.

Ranking	Card	Sub	Goal
Top-1	1.39	-2.69	10.84
Top-2	3.02	-1.40	11.81
Top-3	2.43	-5.47	11.60
Top-4	1.63	-4.52	8.86
Top-5	-0.02	-2.73	15.35

Table 2 shows the differences between the time of largest attention and the annotated frame of each action. Note that the ranking in this table is the average of the differences between the n-th largest attention score and the annotated frame for each label over all samples. In Table 2, we discover that the interesting time of each action varies. For example, the card scene attention is highest around the same or shortly after the annotated frame, focusing on scenes when the referee holds up the card, see Fig. 2(a) and (d). For player substitution, the highest attention score is before the annotated frame. This is because the model focuses on the board for player change before the annotated label frame, see Fig. 2(b) and (e). For goal scenes, the highest attention is on frames a few seconds after the goal shot itself. This is because our model focused on replay and player actions, shown in Fig. 2(c) and (f).

5 Conclusion

In this paper, we proposed a method for action spotting with a transformer to capture related scenes for actions in soccer videos. We demonstrate the effectiveness of the method on SoccerNet, where it outperforms previous work on action spotting. Furthermore, we show that temporal attention is able to highlight discriminative features in the temporal neighborhood of each action. An avenue for future work is extending the application scope of the model to more action classes and more general types of sports and action videos.

References

[1] Gu Chunhui, Sun Chen, A Ross David, Vondrick Carl, Pantofaru Caroline, Li Yeqing, Vijayanarasimhan Sudheendra, Toderici George, Ricco Susanna, Sukthankar Rahul, Schmid Cordelia, and Malik Jitendra. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Computer Vision and Pattern Recognition*, pages 6047–6056, 2018.

[2] A. Sigurdsson Gunnar, Varol Gül, Wang Xiaolong, Farhadi Ali, Laptev Ivan, and Guptak Abhinav. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526, 2016.

[3] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *European Conference on Computer Vision*, pages 392–405, 2010.

[4] Bettadapura Vinay, Pantofaru Caroline, and Essa Irfan. Leveraging contextual cues for generating basketball highlights. In *Proceedings of ACM Conference on Multimedia*, 2016.

[5] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. In *arXiv:1609.08675*, 2016.

[6] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[7] Silvio Giancola, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. Soccernet: A scalable dataset for action spotting in soccer videos. In *Computer Vision and Pattern Recognition Workshops*, June 2018.

[8] Vignesh Ramanathan, Jonathan Huang, Sami Abu-El-Haija, Alexander Gorban, Kevin Murphy, and Li Fei-Fei. Detecting events and key actors in multi-person videos. In *Computer Vision and Pattern Recognition*, pages 3043–3053, 2016.

[9] `https://www.fifa.com/worldcup/archive/russia2018/statistics/`. accessed: March 29, 2021.

[10] Anthony Cioppa, Adrien Deliège, Silvio Giancola, Bernard Ghanem, Marc Van Droogenbroeck, Rikke Gade, and Thomas B. Moeslund. A context-aware loss function for action spotting in soccer videos. In *Computer Vision and Pattern Recognition*, pages 13126–13136, 2020.

[11] Matteo Tomei, Lorenzo Baraldi, Simone Calderara, Simone Bronzin, and Rita Cucchiara. Rms-net: Regression and masking for soccer event spotting. In *arXiv:2102.07624*, 2021.

[12] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, 2014.

[13] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, 2015.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[15] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, volume 27, pages 569–576, 2014.

[16] Caba Heilbron Fabian, Escorcia Victor, Ghanem Bernard, and Carlos Niebles Juan. Activitynet: A large-scale video benchmark for human activity understanding. In *Computer Vision and Pattern Recognition*, pages 961–970, 2015.

[17] Mohammadreza Zolfaghari, Gabriel L. Oliveira, Nima Sedaghat, and Thomas Brox. Chained multi-stream networks exploiting pose, motion, and appearance for action classification and detection. In *International Conference on Computer Vision*, pages 2904–2913, 2017.

[18] Pan Junting, Chen Siyu, Shou Zheng, Shao Jing, and Li Hongsheng. Actor-context-actor relation network for spatio-temporal action localization. In *arXiv:2006.07976*, 2020.

[19] Vicky Kalogeiton, Philippe Weinzaepfel, Vittorio Ferrari, and Cordelia Schmid. Action tubelet detector for spatio-temporal action localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4405–4413, 2017.

[20] Georgia Gkioxari and Jitendra Malik. Finding action tubes. In *Computer Vision and Pattern Recognition*, pages 759–768, 2015.

[21] Tran Du, D Bourdev Lubomir, Fergus Rob, Torresani Lorenzo, and Paluri Manohar. C3d: generic features for video analysis. In *arXiv:1412.0767*, 2014.

[22] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. Long-term recurrent convolutional networks for visual recognition and description. In *Computer Vision and Pattern Recognition*, pages 2625–2634, 2015.

[23] Miura Koichi and Hamada Reiko. Motion based automatic abstraction of cooking videos. In *Computer Vision and Image Media*, pages 21–29, 2003.

[24] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. In *Computer Vision and Pattern Recognition Workshops*, 2017.

[25] Tang Peng, Wang Xinggang, Shi Baoguang, Bai Xiang, Liu Wenyu, and Tu Zhuowen. Deep fishernet for image classification. In *Transactions on Neural Networks and Learning Systems*, pages 2244–2250, 2019.

[26] Philbin James, Chum Ondrej, Isard Michael, Sivic Josef, and Zisserman Andrew. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[27] Girdhar Rohit, Ramanan Deva, Gupta Abhinav, Sivic Josef, and Russell Bryan. ActionVLAD: Learning spatio-temporal aggregation for action classification. In *Computer Vision and Pattern Recognition*, pages 971–980, 2017.

[28] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. In *arXiv:1906.07155*,

2019.

[29] William McNally, Kanav Vats, Tyler Pinto, Chris Dulhanty, John McPhee, and Alexander Wong. Golfdb: A video database for golf swing sequencing. In *Computer Vision and Pattern Recognition Workshops*, 2019.

[30] Dosovitskiy Alexey, Beyer Lucas, Kolesnikov Alexander, Weissenborn Dirk, Zhai Xiaohua, Unterthiner Thomas, Dehghani Mostafa, Minderer Matthias, Heigold Georg, Gelly Sylvain, Uszkoreit Jakob, and Houlsby Neil. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[31] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Association for Computational Linguistics*, pages 4190–4197, 2020.

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[33] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[34] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[35] Arandjelovic Relja, Gronat Petr, Torii Akihiko, Pajdla Tomas, Josef, and Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Computer Vision and Pattern Recognition*, pages 5297–5307, 2016.