Soccernet-V3 Player Re-Identification using Body Part Appearences

Mahesh Bhosale
Dept. of CSE
University at Buffalo,NY
mbhosale@buffalo.edu
50418912

Abhishek Kumar
Dept. of CSE
University at Buffalo,NY
akumar58@buffalo.edu
50419133

Abstract — We propose a neural network architecture that learns body part appearances for soccer player re-identification. Our model consists of a two-stream network (one stream for appearance map extraction and the other one for body part map extraction) and a bilinear-pooling layer that generates and spatially pools the body part map. Each local feature of the body part map is obtained by a bilinear mapping of the corresponding local appearance and body part descriptors. Our novel representation yields a robust image matching feature map, which is the result of combining the local similarities of the relevant body parts with the weighted appearance similarity. Our model does not require any part annotation on the Soccernet re-identification dataset to train the network. Instead, we use a sub-network of an existing pose estimation network (OpenPose) to initialize the part sub-stream, then train the entire network to minimize the triplet loss. The appearance stream is pretrained on ImageNet dataset and the part stream is trained from scratch for the soccernet dataset. We demonstrate the validity of our model by showing that it outperforms state-of-the-art models such as Osnet and InceptionNet.

Keywords— Player re-identification, Soccernet, Body Part features

I. INTRODUCTION

The goal of person-re identification is to retrieve the images from a gallery set given an anchor image. The images in the gallery set and the anchor image are generally taken from multiple cameras from different views which renders the task of person re identification challenging. Multiple camera views pose a challenge because the views are disjoint, temporal distance between images is not constant, lighting conditions and backgrounds are different. The challenge of person re-identification for players in any sports is even more challenging as interclass distance is very small because of the high appearance similarity which makes it hard to identify players even to a naked eye. Per-class samples are also very less which renders it even harder. *Fig. 1* shows the player identity association between multiple camera views with varied image size and background.

Person reidentification is necessary in many applications in video surveillance. Player reidentification has important applications in the sports analytics industry. One of the first steps in player tracking task is player reidentification. It is also readily being used in automatic highlight generation and video assistant referee.

Current methods of player reidentification mainly focus on two ways - one is to get high quality discriminative features [1, 2, 3] and other is to define the distance metric which can be used as a loss for learning task [4, 5, 6].

Many methods propose to use multi-scale features [7][8] due to their importance in the task of re-identification. But many of these methods do not work well because of the increased challenges in player reidentification. Due to high similarity in appearance in players due to similar physical and jersey appearance features are not enough to discriminate the dissimilar players from each other. Pose features should therefore also be considered as discriminative cues in addition to the appearance features. In particular, we use the body part features which are learnt using a subnetwork from OpenPose which, coupled with appearance features are able to address the misalignment in body parts and contribute to positional features.

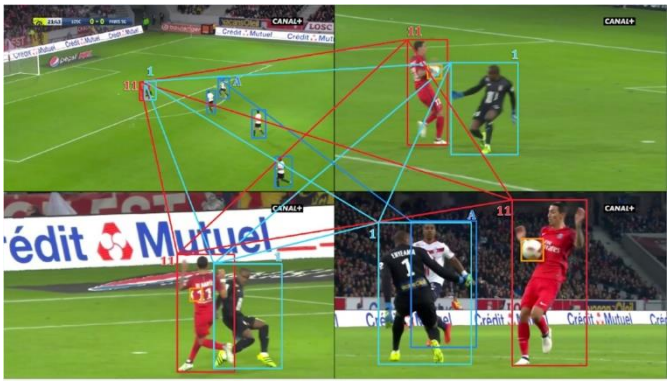

Figure 1Assocation in Soccer-Player Re-identification

II. METHODS

We propose to use spatial body part features in accordance with their appearance to give us rich feature representations. We propose a two-stream network architecture - one stream works on extracting the global appearance features of the

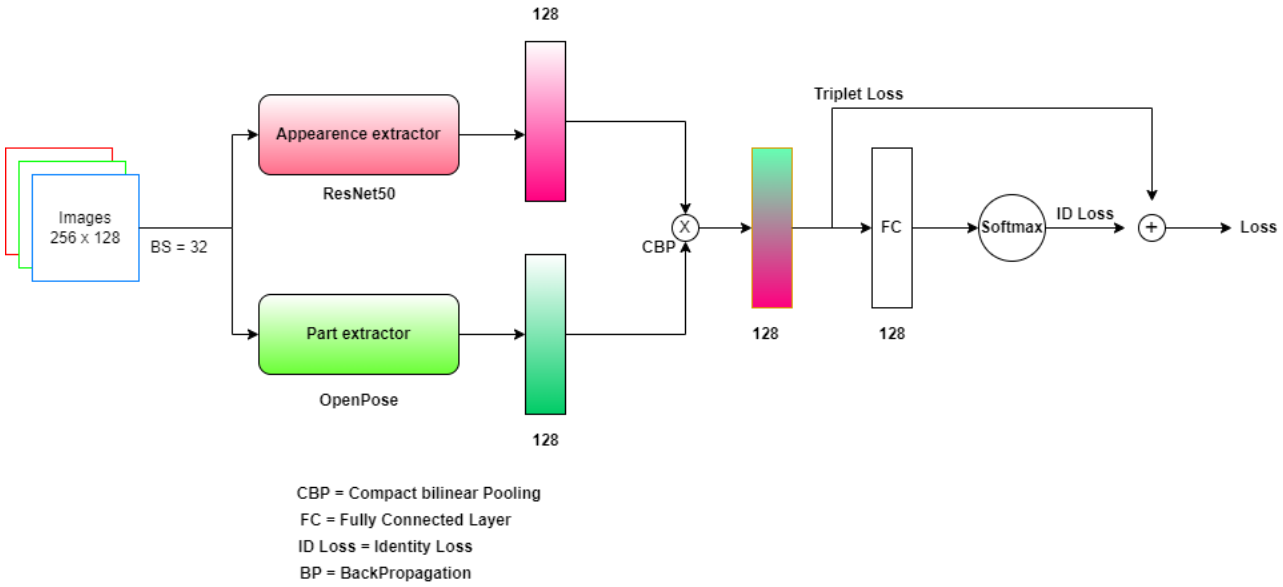

Figure 2 Architecture of proposed two stream network

image whilst the other stream works on extracting the body part features. Features of the two streams are combined with bilinear pooling which allows better interaction between two features giving a high-quality feature representation which captures the appearance of body part features. *Fig. 2* contains the architecture of the proposed model.

A. *Appearance extractor*

Appearance extractor works on extracting the appearance features of the image. We train the RESNET-50 model for this task which is initialized with pretrained weights learnt on the classification task of the Zoo dataset [11]. It takes the input image and outputs the appearance features a, $a \in H \times W$.

B. *Part extractor*

Part extractor works on extracting the spatial features of body parts in an image. It takes in an input image and outputs the body part features $p, p \in H \times W$.

We do not need any annotations of body parts for this task as we train the subnetwork of OpenPose [10] which is initialized with pretrained weights on the task of pose estimation on COCO dataset.

OpenPose is a state-of-the-art method for pose estimation. It is a multi-stage CNN with earlier stages working on capturing the Part Affinity Fields (PAF) which is a 2d vector field which captures the orientation and spatial location of the body parts relative to the image domain. Later stages of CNN take PAF and produce confidence maps of 17 key points of the body parts which are then associated using bipartite graph matching to estimate the pose.

Formally, input to the first stage CNN of OpenPose is a feature map which outputs the first PAF. First PAF with original feature map is given as input to next stage CNN producing second PAF, similarly the outputs of the previous stages are given as input to next stages to produce the final PAF. T_p denotes the number of CNN stages producing part affinity field.

$$\mathbf{L}^1 = \phi^1(\mathbf{F})$$
$$\mathbf{L}^t = \phi^t(\mathbf{F}, \mathbf{L}^{t-1}), \forall 2 \leq t \leq T_P$$

Final PAF is given as input to the first stage CNN working on a confidence map which outputs the first part confidence map. Similar to earlier stages of OpenPose working to produce PAF, output of previous stages with the original feature map is recursively fed as input to the next stages, which finally produces the final confidence map. T_c denotes the number of CNN stages producing part affinity confidence map.

$$\mathbf{S}^{T_P} = \rho^t(\mathbf{F}, \mathbf{L}^{T_P}), \forall t = T_P$$
$$\mathbf{S}^t = \rho^t(\mathbf{F}, \mathbf{L}^{T_P}, \mathbf{S}^{t-1}), \forall T_P < t \leq T_P + T_C$$

We use the final part affinity confidence map St of the last CNN stage.

Fig. 3 describes the multi-stage architecture of OpenPose, earlier stages working on PAF while later stages working on Part affinity confidence maps.

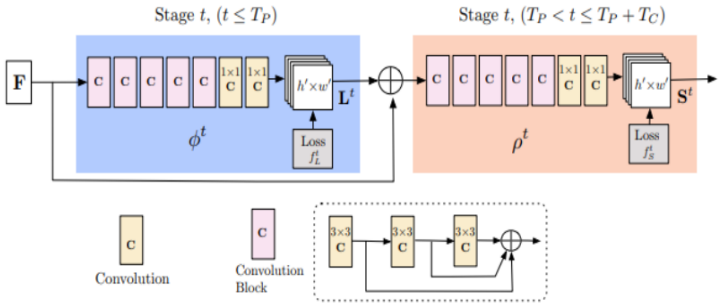

Figure 3 OpenPose Multi-Stage CNN architecture

Readers are strongly advised to read the OpenPose [10] research paper for a more detailed overview of how OpenPose works.

C. *Bilinear Pooling*

Bilinear pooling originated as a method of feature extraction for fine-grained visual recognition where authors introduced Bilinear CNNs [14]. It was used to classify the categories of a bird, where one stream was trained to extract the part features of a bird while the other stream was trained to extract the texture features. Similarly, we bi-pool the features from both the streams, which simply means taking the outer product to allow finer interactions of the features.

Formally bilinear pooling is defined as,

$$\mathbf{f} = \text{pooling}_{xy}\{\mathbf{f}_{xy}\} = \frac{1}{S}\sum_{xy}\mathbf{f}_{xy}$$

where, $\mathbf{f}_{xy} = \text{vec}(\mathbf{a}_{xy} \otimes \mathbf{p}_{xy})$

x and y are spatial locations in the image and S is spatial size. a_{xy} is appearance feature at location (x, y) while p_{xy} is part feature at location (x, y). The pooling operation we use here is average pooling. $vec(.)$ transforms a matrix to a vector, and $\otimes$ represents the outer product of two vectors, with the output being a matrix. Pooled feature therefore incorporates the appearance of body parts in an image, and is therefore part-aligned. Part-aligned feature is normalized at the end, $\tilde{\mathbf{f}} = \frac{\mathbf{f}}{\|\mathbf{f}\|_2}$.

Both the network streams can be trained end-to-end [12], below is a computational graph of two streams A and B with loss l, showing backpropagation in progress.

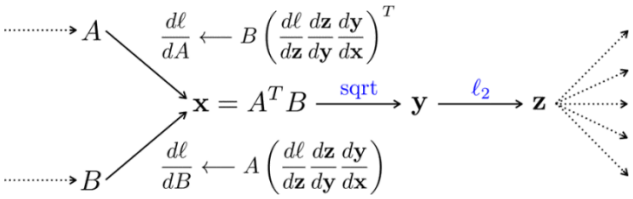

Figure 4 Back propagation in Bi-pooling layer

Compact Bilinear Pooling - To make the bilinear pooling further efficient we use the method of compact bilinear pooling. It reduces the dimension of the vector by taking random projections to approximate the outer product which can be computationally expensive in higher dimensions. We use a method of tensor sketch for projections described in [13], the reader is highly advised to take a look at [13] for detailed explanation.

D. *Layer – Wise similarity*

In deep CNNs, lower layers can identify some low-level features such as shape and edges while upper layers capture the high-level features such as semantic information of an image. Figure 3 shows the activation maps of multiple layers of RESNET-50, starting from layer-1 at the top to layer-6 at the end.

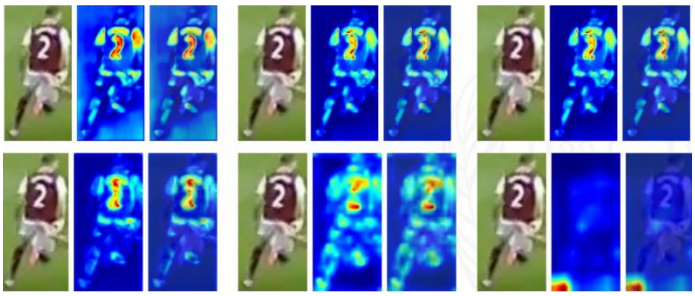

Figure 5 Activation Maps in RESNET-50 (top-left first layer to bottom-right last layer)

As can be seen from the activations maps in *Fig. 5*, activations of the last layer do not reveal much information, so we are also interested in the features extracted at some previous layers of RESNET-50.

Fig. 6 shows the working of proposed Layer-wise similarity method. We take the output features of some of the hidden layers, add fully connected layers to enable better interactions between features and calculate the metric learning loss on these feature vectors. We generally used similarity loss described in the next section, and since we calculate it at multiple layers it is called layer wise similarity loss. Number of layers, choice of layers and number of fully connected layers is tuned in the process of cross validation. This adds supervision which requires feature maps of two similar images to be similar (with the use of triplet loss) in hidden layers of CNN promoting better learning. Results section shows that adding the layer-wise similarity is useful in increasing the mAP and Rank-1 scores.

E. *Optimization Objectives*

Person reidentification can be considered as a task of image retrieval, therefore the image of a player under consideration is called as anchor image a (taken from the query set as described in dataset section) and the image of the same player for same action (similar timeframe) but from other camera view (taken from the gallery set as described in dataset section) is called as positive image p whilst the image of different player for the same action (similar timeframe) from different or same view (taken from the gallery set as described in the dataset section) is called as negative image n. Triplet loss is then defined as,

$$L = \max(d(a,p) - d(a,n) + margin, 0)$$

where d can be any distance metric such as L1/L2 distance. Here we use L2 distance. Triplet loss therefore tries to pull two similar images (a, p) together while pushing apart the two dissimilar images (a, n). This also is generally called similarity loss.

We can also model the problem of player reidentification as a task of classification, therefore we can also use cross entropy loss, which is formally defined as,

$$L = -\frac{1}{m}\sum_{i=1}^{m} y_i \cdot \log(\hat{y}_i)$$

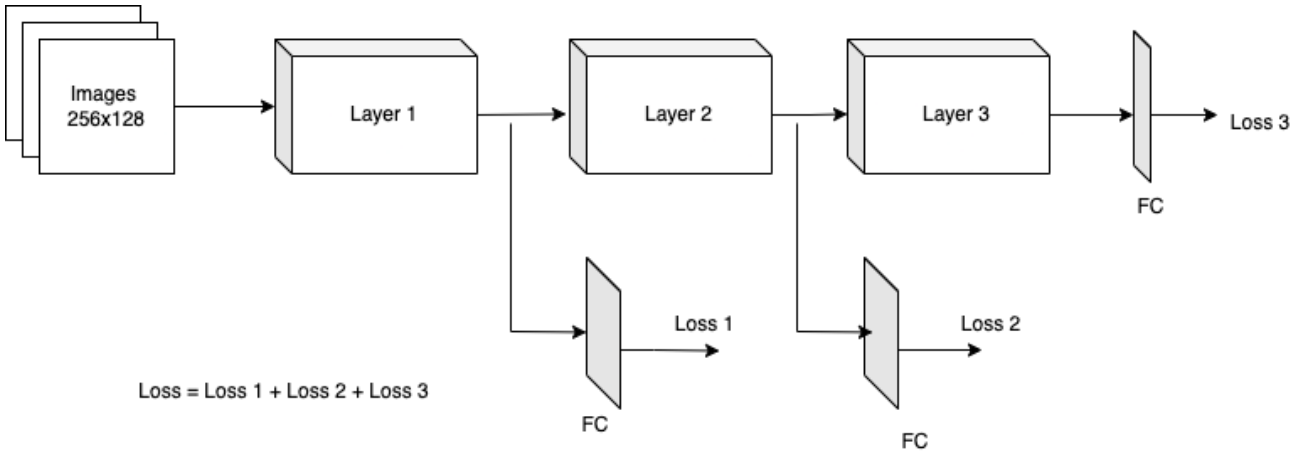

Figure 6 Proposed Layer Wise Similarity architecture

y_i is ground truth pid whilst $\hat{y}_i$ is predicted pid. It is also called identity loss.

III. Framework for Evaluation

A. Dataset

Soccernet-V3: We implement our model on the Soccernet-V3 dataset. The Soccernet-V3 Re-Identification (ReID) dataset is composed of 340.993 players thumbnails extracted from image frames of broadcast videos from 400 soccer games within 6 major leagues. Soccer players from the same team have a strikingly similar appearance, making it difficult to identify them apart in the SoccerNet-v3 ReID dataset. Image resolution also varies significantly, and each identity has a limited number of samples, making the model more difficult to train. This makes the soccernet-V3 dataset more challenging.

The soccernet-V3 dataset is divided into Train/valid/test/challenge sets. The train set is used for training the model. Player identification labels are formed from links between bounding boxes within an action, they can only be used within that action. As a result, player identity labels do not persist between actions, and each action in which a player has been seen has a separate identity. As a result, only samples from the same action are matched against each other during the evaluation process.

Image filename convention:

- Filename convention in the *train/valid/test* sets:

<bbox_idx>_<action_idx>_<person_uid>_<frame_idx>_<class>_<ID>_<UAI>_<height>x<width>.png

e.g. '54404-1940-33397-5089-Player_team_right-9-014r002_00858c3b0003-775x449.png'

- Filename convention in *challenge* set

Annotations for the challenge set are kept secret, the filename convention for challenge set samples is therefore:

<bbox_idx>_<action_idx>_<height>x<width>.png

In *Fig. 7* we can see images of a player in the same action, from different viewpoints in a and b having the same pid. In c we can the same player in different action with different pid.

Figure 7 Same Player Image from different actions left to right a). b) c)

B. Metrics and Evaluation

Each valid/test/challenge set is divided into two subsets similar to typical ReID datasets: query and gallery. The query contains images from the action frames and the gallery contains images from the replay frames. The bounding boxes from action frames with at least one match in the replay frames are used as query samples. Bounding boxes from replay frames or action frames with no match make up the gallery samples. We compute a ranking of gallery samples from the same action for each query, as well as the ranking performance: rank-1 and mean average precision. As a result, query samples are only matched with gallery samples from the same action. Query samples are only matched to gallery samples from distinct camera views in traditional street surveillance ReID databases.

We use Rank-1 accuracy and mAP as evaluation metrics.

IV. Results and Discussion

In the training phase we used a batch size of 32 and trained the model on 10% of the dataset. We optimized our triplet loss function using Adam optimizer with a learning rate of 0. 00001.The input image resolution is kept as 256x128. For comparison the settings and hyperparameters are kept similar for all evaluation done using different baselines on the dataset.

The tables show that our model can outperform the alternative baselines. In particular we were able to surpass the Osnet model, which is the current state of the art for soccer player

Sr no.	Model	mAP	Rank-1
1	Our	63.7	52.8
2	Osnet	61.6	51.2
3	inceptionetv4	46.7	32
4	Resnet50mid	46.5	31.7
5	Resnet50_baseline	46.7	32.8

Table 1 : Evaluation results on 10% data

Sr no.	Model	mAP	Rank-1
1	Osnet	55.5	45.1
2	Our	55	42.4
3	inceptionetv4	49.9	35.8
4	Resnet50mid	42.9	27.8
5	Resnet50_baseline	44.2	28.9

Table 2 : Evaluation results on 2% data

reidentification by 2.1% on mAP when training the model on a 10% dataset. The results give a strong indication that our two stream neural network architecture has a great potential for identifying features of heterogeneous scales and viewpoints,

and thus, should be considered for a broad range of visual recognition tasks.

Below is the output of our layer-wise similarity model. We can see that by using similarity scores from intermediate levels we were able to increase our mAP by 3.7% and Rank-1 score by 3.6%. The layer-wise similarity model was trained on 10%.

Sr no	Model	mAP	Rank-1
1	Resnet_baseline	46.7	32.8
2	Our_layerwise_similarit	50.4	36.2

Table 3 : Evaluation results with Layer-wise similarity

V. CONCLUSION

We proposed a two-stream neural network architecture for soccer player reidentification. The key elements of our model are (1) appearance feature and (2) body part feature from the two-stream network and (3) a compact bilinear pooling method to fuse the two feature maps and generate a rich body part appearance map. Extensive experiments using our model on the soccernet-V3 dataset demonstrated that the two-stream network was able to re-identify soccer players with high mean average precision and Rank-1 accuracy. We also saw that low level features had rich semantic information and we can further enhance our model by considering these low-level features in calculating layer wise similarity scores and optimize the triplet loss as future work.

VI. REFERENCES

[1] F. Xiong, M. Gou, O. Camps, and M. Sznaier. Person reidentification using kernel-based metric learning methods. In the European Conference on Computer Vision (ECCV), September 2014.

[2] R. Zhao, W. Ouyang, and X. Wang. Person re-identification by salience matching. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2013

[3] W. Li and X. Wang. Locally aligned feature transforms across views. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2013.

[4] D. Chen, Z. Yuan, G. Hua, N. Zheng, and J. Wang. Similarity learning on an explicit polynomial kernel feature map for person re-identification. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.

[5] J. Chen, Z. Zhang, and Y. Wang. Relevance metric learning for person re-identification by exploiting global similarities. In The IEEE International Conference on Pattern Recognition (ICPR), August 2014.

[6] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2009.

[7] Kaiyang Zhou et.al, Omni-Scale Feature Learning for Person Re-Identification, in International Conference on Computer Vision (ICCV), 2019.

[8] Yanbei Chen et.al, Person Re-identification by Deep Learning Multi-scale Representations, in International Conference on Computer Vision workshops (ICCV workshop), 2017.

[9] Yulin Gao et.al., Efficient and Deep Person Re-Identification using Multi-Level Similarity in CVPR (2018)

[10] Zhe Cao et.al, OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields in IEEE transactions on pattern analysis and machine intelligence (2019)

[11] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository , Zoo dataset. Irvine, CA: University of California, School of Information and Computer Science.

[12] Tsung Yu lin et.al, Bilinear CNN Models for Fine-Grained Visual Recognition, International Conference on Computer Vision (ICCV) 2015.

[13] Yang Gao et.al, Compact Bilinear Pooling, in Conference on Computer Vision and Pattern Recognition (CVPR) 2016.