Transductive Weakly-Supervised Player Detection using Soccer Broadcast Videos

Chris Andrew Gadde[1] C.V. Jawahar[2]
International Institute of Information Technology, Hyderabad, India
[1]chris.andrew@research.iiit.ac.in, [2]jawahar@iiit.ac.in

Abstract

Player detection lays the foundation for many applications in the field of sports analytics including player recognition, player tracking, and activity detection. In this work, we study player detection in continuous long shot broadcast videos. Broadcast match videos are easy to obtain, and detection on these videos is much more challenging. We propose a transductive approach for player detection that treats it as a domain adaptation problem. We show that instance-level domain labels are significant for sufficient adaptation in the case of soccer broadcast videos. An efficient multi-model greedy labelling scheme based on visual features is proposed to annotate domain labels on bounding box predictions made by our inductive model. We use reliable instances from the inductive model inferences to train a transductive copy of the model. We create and release a fully annotated player detection dataset comprising soccer broadcast videos from the FIFA 2018 World Cup matches to evaluate our method. Our method shows significant improvements in player detection to the baseline and existing state-of-the-art methods on our dataset. We show, on average, a 16 point improvement in mAP for soccer broadcast videos by annotating domain labels for around a 100 samples per video.

1. Introduction

Analysis of sports broadcast videos is an exciting and relatively unexplored area of research in computer vision and media understanding. Tasks such as event detection, activity recognition, player tracking, and team analysis are helpful applications to understand and analyze a game. These downstream task require player detection as their primary basis or as supplementary information [8, 19, 11]. Transductive approaches for player detection are interesting because of the large inter-match variability in soccer videos. Models trained on one match, tend to perform poorly on other matches, showing significant domain-shift between different matches. Some amount of fine-tuning of models is needed in most cases, however, labelled data for fine-tuning is rarely available. Other challenges like player occlusion, pose variations, player truncation, and motion blurring also exist but can be rectified with sufficient amount of training data. A transductive approach uses unlabelled target data for improving detection performance for a given match which can be easily applied to improve performance in new matches for which labels are not available.

(a) Team managers & support staff **(b)** Cameramen

(c) Referees **(d)** Audience members

Figure 1: Domain-noise present in soccer broadcast videos. Detections are correct for the *person* class, however there are many false positives for the *player* class. This noise prevents using transductive or self-supervised approaches for training, without being removed.

Existing player detection approaches utilise input from the camera feed or using a single view of the match for detection [18, 32, 10, 14] which is not readily available. Broadcast videos of the matches are a much more accessible source of unlabelled information. However, there are many challenges in broadcast video data as compared to fixed or constrained camera views. Our work addresses the major problem of *domain-noise* in broadcast videos, as shown in Fig 1. We show that instance-level *domain labels* are needed in broadcast videos to achieve good performance

with a transductive approach using unlabelled target data.

A pre-trained inductive model is used to get an initial set of bounding boxes for a video. The bounding boxes are clustered using visual information extracted from a person re-identification model. A novel multi-model greedy clustering approach is introduced to assign instance level *domain labels* to the bounding boxes collectively. We then use *reliable* instances from our initial proposals to update model parameters of a transductive copy of our detection model and show a significant improvement in player detection performance. We achieve, on average, an 16 point performance improvement in mAP by annotating around a 100 samples with *domain labels* per video. We also find a lack of publicly available datasets for continuous long shot broadcast videos. Therefore, to evaluate the performance of our approach, we create and release a fully annotated player detection dataset comprising of broadcast videos from the FIFA 2018 World Cup matches. This dataset also serves as useful supplementary information for other downstream tasks in sports analysis.

The main contributions of the paper are as follows: (i) we propose a novel transductive approach to player detection which achieves significant performance improvements using only a few domain labelled samples; (ii) we introduce an approach to collectively assign *domain labels* at the instance-level, which is necessary for target data such as broadcast videos that contain a lot of *domain-noise*; (iii) we create and release a fully annotated dataset consisting of soccer broadcast videos to evaluate the task of player detection.

2. Related Works

Player Detection: Most player detection systems rely on setting up specific equipment beforehand to process the live match and rarely deal with broadcast videos [1, 29, 20]. Earlier works such as [18, 32, 11] do not utilise modern deep learning based object detection systems and place constraints on the camera view of the input video. Recent works like [10] utilise a student-teacher training paradigm to learn a smaller network using an improved teacher network that learns missed detections on the training data through a blob detection strategy and human annotations. The results are based on wide-angle fixed camera views of matches where the number of false positives is low, and fine-tuned detectors tend to perform well. In [14], the authors use a Feature Pyramid Network and combine lower level features with higher spatial and higher level features with a bigger receptive field to train a small network that performs well on the ISSIA [4] and Soccer Player Detection dataset [17]. The authors propose a supervised approach for detection and require annotated training data to be available.

Player Detection Datasets: The ISSIA soccer dataset [4] consists of 18000 frames annotated with player and referee bounding boxes. The data includes a single match seen from 3 different wide-angle camera views. In contrast, broadcast videos contain multiple shots at different zoom levels, camera movements, and shot transitions. In [17], the authors propose a player detection dataset that consists of 2019 annotated images recorded by three broadcast pan-tilt-zoom (PTZ) cameras. The dataset contains too few samples to be exhaustive enough to cover all different possible scenarios or train modern detection architectures. The SoccerDB dataset [13] consists of 346 clips from the SoccerNet dataset [7] annotated for player detection using an automated labelling scheme. The videos are sourced from broadcast matches but each clip consists of only a few hundred frames and lacks sufficient shot transitions in the clips. Since we see a lack of continuous long-shot broadcast videos in the publicly available datasets, we create and release a dataset of our own for evaluation, discussed in detail in Section 5.

Domain Adaptive Object Detection: One of the first works on domain adaptation for object detection was shown in [2]; where the authors use domain classifiers and consistency regularization to improve object detection performance on the KITTI dataset [6]. In [27], the authors use distillation loss along with soft labels to perform single-category detection tasks like face and pedestrian detection. The soft labels are generated with the help of a tracking algorithm. Only one work has been done using a transductive approach to detect objects which is shown in [23]. The authors propose a zero-shot learning paradigm using fixed and dynamic pseudo-labels to train a transductive model that performs better on the target domain. They also incorporate semantic information using word vectors to generate the labelling. Such additional sources of information to improve the performance may not be available for broadcast videos. Most of these works assume that domain labels exist at the image-level and are the same at the instance-level. However, this assumption does not hold true for soccer broadcast videos and is discussed in Section 3. We also demonstrate how to overcome this problem using a greedy multi-model collective labelling approach.

3. Domain Noise

We establish a distinction between a *person* and a *player*, in that we only consider the 22 players comprised of the two teams in a soccer match as valid instances. We observe poor performance of pre-trained detectors like [24, 25] for the *player* class on broadcast videos compared to the performance on the *person* class on large-scale datasets like [16]. This performance gap is due to many false positives, such as members of the audience, referees, and camera operators observed in the qualitative analysis of pre-trained detectors as shown in Fig. 1. These instances serve as *domain noise* during the domain adaptation process, specifically for unsu-

pervised approaches.

We define *object detection* as learning the joint distribution $P(C, B, I)$. Where C denotes the class label, B denotes the bounding box and I denotes the input image. As shown in [2], the joint distribution can be effectively decomposed as in Eq. 1.

$$P(C, B, I) = P(C|B, I)\, P(B, I) \tag{1}$$

The bounding box classifier denoted by $P(C|B, I)$ is assumed to be consistent across domains and the domain shift occurs in this case due to the detector represented by $P(B, I)$, which is further decomposed as $P(B, I) = P(B|I)\, P(I)$. To rectify this domain shift, one must jointly consider image-level and instance-level domain adaptation, such that $P(I)$ and $P(B|I)$, respectively, are consistent across the source and target domain. In the usual setting, image-level and instance-level adaptation is done using samples labelled to be from the source or target domain. Several approaches for domain adaptation make use of this *domain label*, specifically, those that use domain classifiers [2, 21, 5]. In most cases, these *domain labels* are annotated at the image-level and are assumed to hold true for all instances in the image. However, this assumption does not hold true for soccer broadcast videos, as shown previously. Without removing invalid instances, we observe that sufficient image-level adaptation can be achieved but instance-level adaptation is lacking. This is true for any real-world data that contains instances mixing in from multiple domains. We therefore propose an efficient framework to assign a domain label to each bounding box instance using a greedy multi-model collective labelling scheme based on visual features. Our application of this framework is used for soccer broadcast videos but the approach can be applied for any real world data where *domain noise* is prevalent. For our approach, we train a transductive model using only *valid* instances from the target domain data available by pruning this noise from the initial predictions made by our inductive model.

4. Our Approach

An overall picture of the approach we have used in this work is illustrated in Figure 2. Transductive approaches for learning have an inductive phase where a pre-trained copy of the detection model(inductive model) generates predictions. We train a better copy of the model(transductive model) for a specific domain or task using these initial predictions. We define ***reliable*** bounding boxes as those with a high confidence level and those identified to belong to the target domain, *i.e.*, *players*. To simplify labelling bounding boxes as *reliable*, we use a greedy multi-model clustering approach to aggregate similar bounding boxes and label them collectively, reducing the annotation load significantly. The predictions identified as *reliable* from the inductive phase are used update the parameters of the transductive model to get better predictions.

4.1. Inductive phase

In the inductive step, we get both the initial predictions made by our detection system and the visual features we use to cluster the bounding boxes together. A YOLOv3 [24] detector with spatial pyramid pooling (SPP) [9] that has been pre-trained on the MS COCO [16] dataset is used as the inductive model. B_I^{ij} are the initial predictions made by inductive model where i is the frame identifier and j is the jth prediction for that frame, as given by Eq. 2.

$$\{B_I^{ij}\} = NMS(f_I(X_i)) \tag{2}$$

f_I represents the forward pass of the model and NMS represents the Non-Maximal Suppression function [24] used to get the bounding boxes from the output of the forward pass of the model.

We use a person re-identification model for obtaining the visual features for the instance level labelling scheme, the architecture is illustrated in Figure 3. We found that many audience members and support staff wear team jerseys in the broadcast match and therefore descriptors from simpler image classification networks were not sufficient to distinguish between them and the players. The model is a wide residual network [33] with one convolutional layer and four residual blocks pre-trained using the MARS dataset [34] following the method described in [31]. The visual features of dimensionality 512 are computed in the output of the average pool layer of the model. We consider x_{ij} to be the cropped region from the image for bounding box B_I^{ij} which is passed through the re-identification model to get visual features as shown in Eq. 3. Where $g(x)$ is the output from the average pool layer of the re-identification model.

$$f_{ij} = \frac{g(x_{ij})}{||g(x_{ij})||_2} \tag{3}$$

4.2. Identifying reliable predictions

We create a similarity graph between bounding boxes obtained in the inductive phase. We then use a greedy approach to perform *cluster deletion* on the bounding boxes based on the similarity graph. We finally, obtain a representative sample for each cluster and label the cluster as *reliable* or *unreliable*.

4.2.1 Similarity Graph:

Before clustering the bounding boxes, we define a similarity graph G, to create the clusters. Nodes of this graph are

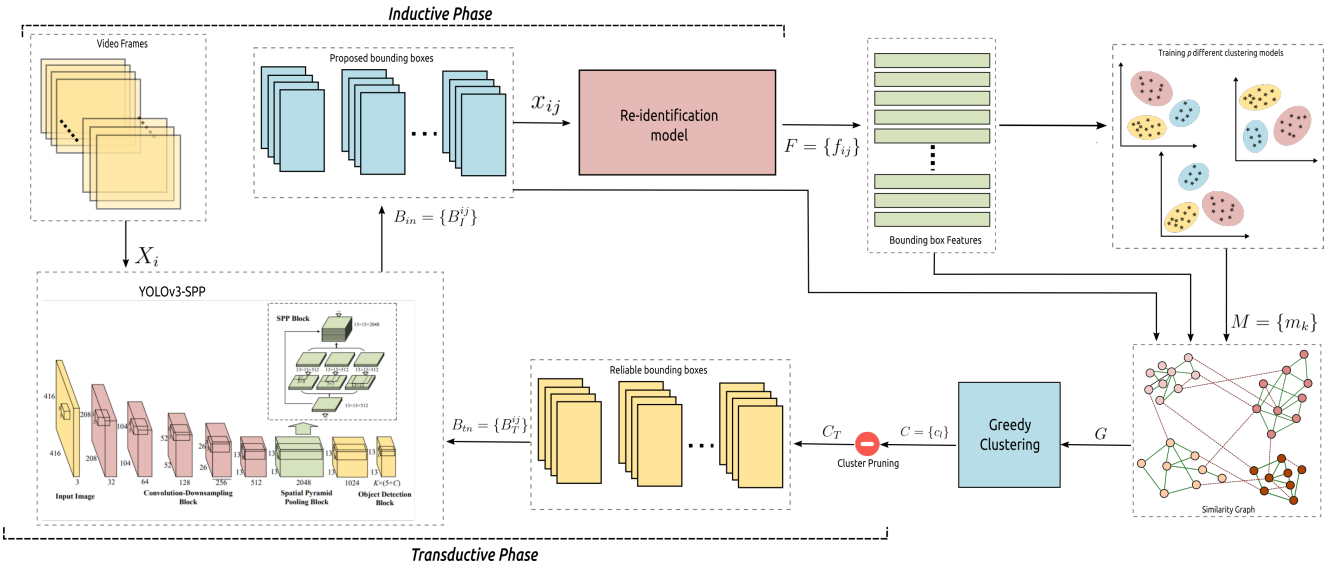


Figure 2: An overview of the detection pipeline proposed. The **inductive** phase includes using a pre-trained detection model to obtain initial bounding box proposals and a re-identification model to obtain visual features. The **second** stage includes clustering the obtained bounding boxes and labelling them as *reliable* or *unreliable* using a multi-model greedy clustering approach. The **transductive** phase includes using *reliable* bounding boxes to fine-tune the detector parameters to perform better detection.

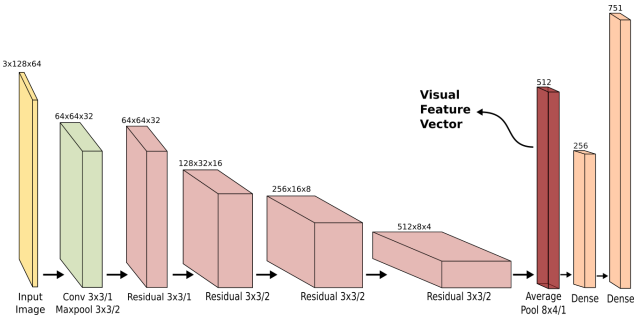


Figure 3: Re-identification model architecture used. The model is a wide residual network [33] consisting of 4 residual blocks.

the bounding boxes obtained in the inductive phase, and the edges of the graph are generated using a multi-model similarity metric. p different unsupervised clustering models are trained using a random subset of visual features f_{ij}. We use a combination of the k-means algorithm and Gaussian Mixture Models [26] with varying clusters, more details in the supplementary. The models are used to define positive and negative edges between the nodes. Edges are created using similarity metrics $S(x, x')$ and $\hat{S}(x, x')$ where x and x' are any two given bounding boxes. $S(x, x')$ denotes the number of clustering models that assign the same clusters to the two given bounding boxes based on their corresponding features in F. Similarly, $\hat{S}(x, x')$ denotes the number of models that assign a different clustering to the two given bounding boxes. We say that two samples have a positive edge if $S(x, x') \geq t_p$ where t_p is a pre-defined threshold. Two samples have a negative edge if $\hat{S}(x, x') \geq t_p$. Since we use the same threshold for both cases, we can select t_p relative to p such that the two cases are mutually exclusive and two bounding boxes can have only a positive, negative, or no edge between them.

4.2.2 Greedy cluster deletion and labelling:

We propose a clustering approach used to cluster the bounding boxes together based on the similarity graph defined in the previous section. Our approach is inspired by the Lambda Correlation Clustering (LambdaCC) algorithm proposed in [30] where *cluster deletion* is defined as the problem of finding a minimum number of edges in a graph to be deleted to convert it into a disjoint set of cliques(clusters). Our approach is a greedy approximation that optimises the LambdaCC [30] objective function defined as:

$$\min \sum_{(i,j)\in E^+} (1-\lambda)x_{ij} + \sum_{(i,j)\in E^-} \lambda(1-x_{ij}) \quad (4)$$

Given a connected graph with positive edges (E^+) and negative edges (E^-), with x_{ij} as binary distances for the edges. A high lambda value gives a large penalty if negative edges are present inside a cluster. This ensures that the clusters are internally dense and externally sparse.

We define sets $F = \{f_{ij}\}$, $B = \{B_I^{ij}\}$ and E^+ and E^- as the positive and negative edges in the similarity graph respectively. Our greedy approximation of LambdaCC [30] for *cluster deletion* is defined in Algorithm 1. The algorithm starts by randomly sampling a bounding box from the

Algorithm 1: Greedy approximation of cluster deletion approach

Output: A set of clusters $C = \{c_l\}$ containing visually similar bounding boxes
Input: $B = \{B_I^{ij}\}$, $F = \{f_{ij}\}$, E^+, E^-, t_s
$C \leftarrow \{\}$ (Initialise an empty set C);
while $|B| \neq 0$ **do**
  Randomly sample bounding box b and the corresponding feature f from B and F;
  $c_l \leftarrow \{b\}$ (Initialise c_l with sample b);
  for $x \in B - \{b\}$ **do**
    if $(x, b) \in E^+$ *and* $\forall x_j \in c_l | (b, x_j) \notin E^-$ **then**
      $B \leftarrow B - \{x\}$ (Remove x from B);
      $c_l \leftarrow c_l \cup \{x\}$ (Add x to c_l);
    end
  end
  if $|c_l| \geq t_s$ **then**
    $C \leftarrow C \cup c_l$ (Add c_l to C);
  end
end
return *Set of clustered bounding boxes* $C = \{c_l\}$

inductive set and growing a cluster around the box using boxes with a positive edge connected to the sampled box. We ensure that no negative edges are present inside the cluster. This ensures homogeneity in the cluster, making them dense internally. In the best case scenario, where $t_p = p$, the algorithm runs in linear time as negative edges are ignored. The trade-off is that a larger number of bounding boxes will end up in single element clusters and leave fewer *reliable* samples for learning in the transductive phase. We also define a threshold on the cluster size called t_s, this removes small clusters that mainly contain outliers which may be inherently *unreliable* predictions.

We select the representative sample using the optimisation mentioned in Eq. 5.

$$r_i = \min_x \left(\sum_{x_j \in c_i} \frac{x_j - x}{|c_i|} \right)^2 \tag{5}$$

r_i is the representative sample for cluster c_i and x_j represents the visual feature used for clustering the bounding boxes for every box in c_i. We proceed to label the obtained representative bounding box corresponding to the visual feature r_i as *reliable* or *unreliable* and propagate the same label across the entire cluster. We collect all the bounding boxes from the set of *reliable* clusters C_T and threshold them based on their prediction confidence with t_c, which we add to the set B_{tn}, and use as our training data in the transductive phase.

4.3. Transductive Step

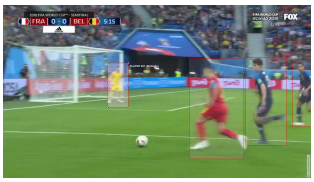
In the transductive step, we create a copy of the inductive model while freezing the backbone, the convolution-downsampling block and SPP block, and fine-tune the remaining layers with *reliable* samples identified in the previous stage. We found that re-training both the detection layers and the convolution upsampling layers of YOLOv3 with SPP yield better performance than training just the detection layers. This helps in the image-level adaptation of the model as mentioned in Section 3, whereas re-training the detection layers help in instance-level adaptation. The backbone of the model has been pre-trained on the ImageNet dataset [3]. We provide training details for this model in the supplementary. The transductive copy of the model is used for player detection and evaluated in Section 6.

5. Dataset

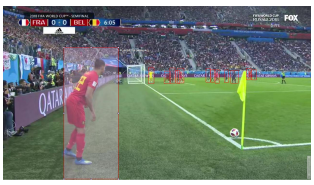
In Section 2, we reviewed the various player detection datasets available in the literature and their suitability to evaluate player detection methods in continuous long shot broadcast videos. Due to the lack of a proper dataset, we have created a comprehensive player detection dataset to evaluate our method. We plan to release this dataset to the public along with this work. Data was collected from the broadcast videos of three FIFA 2018 World Cup matches. The videos are continuous, live, unedited broadcasts of the match telecast to viewers. Annotations for this dataset contain bounding boxes annotated at a per frame level for each video. We have ensured that the videos contain samples of at least four different camera views: top-zoomed-in, top-zoomed-out, bottom-zoomed-in, bottom-zoomed-out, and instances of transition between these views, as shown in Fig. 4. The videos comprise matches between 4 different teams(France, Croatia, Belgium, and England), which ensures variability in player appearances and jersey colors. The dataset consists of 265,625 frame images consisting of 2,115,496 annotated bounding boxes and to the best of our knowledge is the largest such dataset created. We have covered a wide range of player positions and orientations with a large variability of bounding box sizes. There are also multiple instances of player occlusion and the player being truncated out of the frame, as shown in Fig. 4.

The dataset was annotated with the help of a pre-trained YOLOv3 [24] detector along with the DeepSORT [31] tracking algorithm, followed by manual corrections by humans. Instances of false positives, namely, detection of referees, audience members, support staff, and any other non-players are removed. There are also instances where the bounding box annotations need to be manually corrected to fit the player better. We do not ensure that the final corrected annotations are accurate to evaluate player tracking. Nevertheless, we do provide tracking annotations along with the

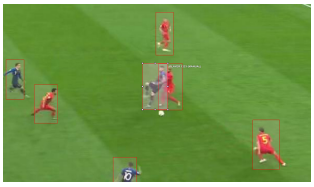
(a) Top camera view



(b) Bottom moving camera view

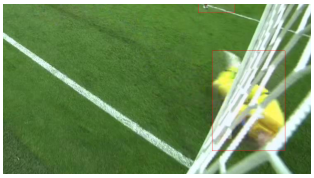


(c) Bottom view with multiple scale bounding boxes



(d) Player pose(kicking)

(e) Player pose(diving)



(f) Partial occlusion with net

Figure 4: Annotated sample images from the dataset, showing different views and scenarios some of which are challenging to perform detection on.

Table 1: Details and statistics of the proposed dataset.

Match	FR vs. CR	FR vs. BE	EN vs. CR
Date	15.07.2018	10.07.2018	11.07.2018
Broadcaster	Fox Sports	Fox Sports	Fox Sports
Resolution	1280×720	1280×720	1280×720
Length(Frames)	95,176	95,944	74,505
Annotated Frames	86,954	89,268	56,096
Total bounding boxes	747,876	950,802	416,818
Avg. bounding boxes	8.6	10.65	7.43
Deleted bounding boxes	90,679	104,933	593,017

detection annotations. We use the CVAT [28] annotation tool for annotating the videos and all annotations are stored in the CVAT annotation format. We provide some useful details and statistics about the dataset and the videos in Table. 1.

6. Results and Experiments

We compare our detection results with pre-trained supervised, fine-tuned and self-supervised approaches for detection. We use precision, recall, and mAP with IoU=0.5 as evaluation metrics, as per the standard practice [22]. We additionally evaluate the performance of our clustering approach compared to other simpler methods in removing unreliable predictions. We also showcase an interesting application of our detector, wherein we use it to generate field heat-maps using top-view registration of player positions on the soccer field.

Table 2: Comparative results on proposed dataset using: pre-trained general purpose detectors; supervised approaches for player detection from SoccerDB [13] and FootAndBall [14]; self supervised approach mentioned in [27] without domain noise removal. Bottom row represent number of samples annotated in the labelling stage.

Method	FR vs. CR			FR vs. BE			EN vs. CR		
	P	R	mAP	P	R	mAP	P	R	mAP
F-RCNN	0.46	0.80	0.38	0.63	0.84	0.54	0.38	0.79	0.38
YOLOv3	0.42	0.83	0.59	0.49	0.85	0.65	0.34	0.82	0.54
RNet	0.50	0.79	0.41	0.62	0.83	0.52	0.40	0.77	0.40
[13]	0.59	0.74	0.45	0.63	0.75	0.48	0.49	0.73	0.40
[14]	0.75	0.62	0.47	0.74	0.67	0.50	0.66	0.53	0.38
[27]	0.31	**0.87**	0.35	0.40	**0.88**	0.51	0.24	**0.86**	0.29
Ours	0.76	0.85	**0.79**	0.89	0.79	**0.76**	0.77	0.78	**0.72**
	105 samples			55 samples			64 samples		

6.1. Baselines

Three widely used detectors, FasterRCNN [25], YOLOv3 [24] with SPP [9](YOLOv3-SPP), and RetinaNet [15], are used as baselines using the *person* class to showcase the poor performance due to false positives in player detection. This showcases the *domain shift* between the two objectives and the need for methods to reduce this. We use pre-trained networks trained on the MS COCO dataset [16]. Results are reported in Table 2.

6.2. Main Results

We observe high detection performance of the transductive model with around 100 samples per video annotated with *domain labels*. We compare our detection results with two recent works SoccerDB [13] and FootAndBall [14], for player detection. Both these works use models trained in a supervised manner on labelled datasets. Results are reported in Table 2. We observe that our method outperforms models trained in a supervised setting by a significant margin since it is trained on target data of the specific videos.

We show some interesting qualitative results in Figure. 5. Images on the left are results generated using SoccerDB [13], and those on the right are results from our model. In the first row, we observe that false positives such as staff wearing the team jersey are not detected. This is a particularly challenging example since team staff wear jerseys similar to players, so distinguishing between them is difficult. In the second row, we observe that no detections are made in our model since no players are present in the frame. This shows that the transductive model has

Figure 5: Qualitative comparison of detection results. *Left*: SoccerDB [13]. *Right*: Our proposed method.

Table 3: Supervised fine-tuning results on YOLOv3[24] with spatial pyramid pooling[9]. The first column shows the percentage of the ground truth data used for training.

Data	FR vs. CR			FR vs. BE			EN vs. CR		
	P	R	mAP	P	R	mAP	P	R	mAP
10%	0.69	0.91	0.85	0.77	0.94	0.89	0.65	0.91	0.83
15%	0.71	0.91	0.86	0.79	0.94	0.90	0.67	0.92	0.84
20%	0.73	0.91	0.86	0.81	0.94	0.90	0.69	0.92	0.85
25%	0.74	0.91	0.86	0.82	0.94	0.91	0.70	0.92	0.85
30%	0.74	0.91	0.87	0.82	0.94	0.91	0.71	0.92	0.86
Ours	**0.76**	0.85	0.79	**0.89**	0.79	0.76	**0.77**	0.78	0.72

Table 4: Comparative results of K-Means, Gaussian Mixture Model(GMM) and our multi-model greedy(MMG) clustering for identifying *reliable* predictions

Video	K-Means		GMM		MMG	
	TPR	FPR	TPR	FPR	TPR	FPR
FR vs. CR	0.97	0.36	0.97	0.45	0.56	**0.82**
FR vs. BE	0.99	0.36	0.96	0.44	0.75	**0.72**
EN vs. CR	0.78	0.53	0.79	0.66	0.74	**0.74**

unlearned the *person* class to a sufficient degree and now focuses only on *players*. In the third row, we observe that the model recognizes players in the initial line up. This is interesting because these detections were not present in the initial predictions made by the inductive model. This indicates that the transductive model has learnt distinct features unique to each player and is therefore making better predictions.

6.3. Comparison with self-supervised approaches

We also compare our work with a recent self-supervised approach that does not utilise instance-level *domain labels* while fine-tuning the model parameters, as shown in Table 2. We train a YOLOv3-SPP detector with the self-training approach mentioned in [27]. We use DeepSORT [31] tracking to generate refined bounding boxes and soft labels, using distillation loss for training as mentioned in [27]. Since there were no *domain labels* at the instance level to isolate the target domain bounding boxes, the model's performance does not improve compared to the baseline. The model also tries to learn the *domain-noise* present in the data. This is highlighted by the fact that recall of the trained model increases while reducing precision, indicating that the model is making more predictions, but it is also generating more false positives.

6.4. Comparison with supervised fine-tuning

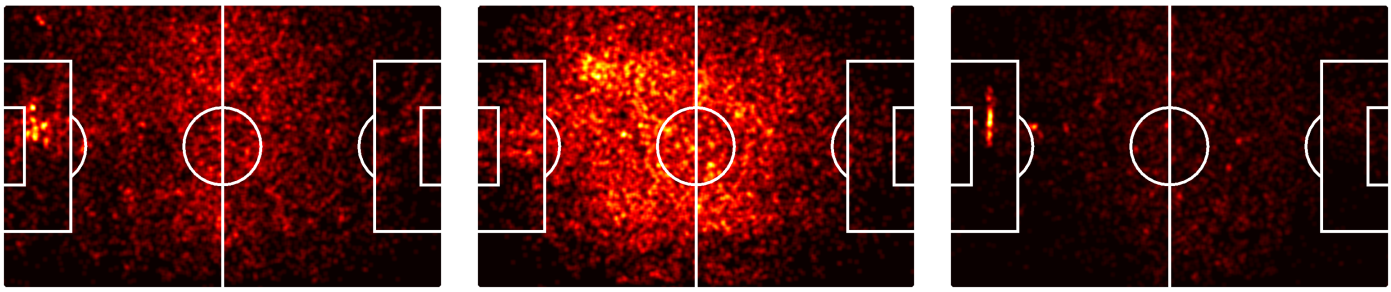
We also evaluate our detection performance against supervised fine-tuning approaches for detection. We randomly sample a percentage of the ground-truth annotations and fine-tune a YOLOv3-SPP model by training the detection and upsampling convolution layer with this data and testing it on the remaining data. Performance is reported in Table. 3. We observe that our results are comparable to those generated by the models trained using ground-truth data while adding only instance-level *domain labels* to around a 100 samples. It is also noted that the precision of our approach is consistently higher, indicating more accurate predictions being made.

6.5. Clustering baselines

We evaluate the performance of our clustering approach alongside simpler clustering approaches using the visual descriptor from our re-identification model for identifying reliable predictions. For evaluation, we introduce two metrics False Positive Removal Ratio(FPR) and True Positive Retention Ratio(TPR). FPR is the ratio of false positives removed to total false positives after pruning using a given clustering method. TPR is the ratio of true positives retained to total true positives after pruning. High values of both these metrics are desirable for successfully identifying reliable predictions. We compare the results of K-Means, Gaussian Mixture Models(GMM), and our multi-model greedy(MMG) clustering in Table 4. We observe that MMG is able to consistently remove more false positives(*domain-noise*) across all videos, while still retaining a significant amount of true positives that are useful for training. Training details are provided in the supplementary.

6.6. Improving image-level adaptation

Experiments are conducted to see which layers in the YOLOv3-SPP need to be trained to achieve better image-level adaptation. In general, while fine-tuning YOLOv3,



(a) France vs. Croatia **(b)** France vs. Belgium **(c)** England vs. Croatia

Figure 6: Heat-map of player locations for first 20 minutes various matches. Brighter regions contain more players.

Table 5: Comparison between training only the detection layers(YOLO) and training both the convolution upsampling and detection layers(YOLO+Up) to achieve better image level adaptation.

Method	FR vs. CR			FR vs. BE			EN vs. CR		
	P	R	mAP	P	R	mAP	P	R	mAP
YOLO	0.64	0.79	0.67	0.69	0.80	0.69	0.66	0.72	0.60
YOLO+Up	0.76	0.85	**0.79**	0.89	0.79	**0.76**	0.77	0.78	**0.72**

only the final detection layers are re-trained using the target domain data. However, we observe that jointly training both the detection and the upsampling convolution layers of our transductive model yielded much richer feature maps at the detection layer. This increases the number of detection the model makes at the NMS stage[24] and improves the detector performance. Comparisons are shown in Table 5.

6.7. Field heat-maps

An interesting application for a reliable detection system is tracking where players are throughout the match. To showcase this, we generate a heat-map of the field using bounding box detections from our transductive model for the first 20 minutes of the FR vs. CR match. We use the method described in [12] for top-view registration of detected bounding boxes in every frame. We then warp the bounding box centers and generate a heat-map over the entire field. We show heat-maps for the first 20 minutes of the three matches in our dataset as shown in Fig. 6. Some interesting inferences can be obtained from these heat-maps about the match. For instance, the heat-map for France vs. Croatia map shows brighter regions towards the left goal post, this goal post represents the Croatia side, which had a goal scored against them at the 18" minute. Thus showing a higher concentration of players on the Croatia side of both defenders and attackers from the opposing team in the first 20 minutes. In France vs. Belgium map, there is a fairly uniform distribution of the players across the field. This is consistent with the match since both teams had a fairly equal possession of the ball and no goal was scored in the video. For England vs. Croatia map, a small region of activity can be seen on the Croatia side(left), where a goal was scored early by England on the 5" minute. Such heat-maps generated per player with reliable detection models and player identities are useful tools in analyzing the game. Heat-maps are just an example of the kind of analysis that is possible with reliable detection systems. It would be interesting to explore further avenues of application for such systems.

7. Conclusion

This work analyzes player detection in unconstrained soccer broadcast videos using a transductive approach for learning without having annotated ground-truth data. We formulate player detection as a domain adaptation problem and highlight the *domain noise* issue that prevents unsupervised and self-supervised forms of learning from performing well. A novel clustering approach is proposed to annotate domain labels collectively at the instance-level to address this noise. We create a dataset comprising soccer broadcast videos to evaluate our method, that we will release publicly. Our model performs player detection better than other supervised and self-supervised methods with only a few samples annotated with domain labels. Using our trained transductive model, we showcase a real world application by generating heat-maps that track player positions across the field and draw inferences about the match from them. Combining accurate detection models with player recognition allows many applications such as tracking the player across the match and searching for actions made by a certain player across the video. Detection models are used by many action detection approaches to localise the region of the frame where an action is being performed. It would be interesting to explore further applications of reliable detection models in future works.

References

[1] Carl Bialik. The people tracking every touch, pass and tackle in the world cup. *Fivethirtyeight. com*, 2014.

[2] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3339–3348, 2018.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[4] Tiziana D'Orazio, Marco Leo, Nicola Mosca, Paolo Spagnolo, and Pier Luigi Mazzeo. A semi-automatic system for ground truth generation of soccer video sequences. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 559–564. IEEE, 2009.

[5] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[6] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[7] Silvio Giancola, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. Soccernet: A scalable dataset for action spotting in soccer videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1711–1721, 2018.

[8] Silvio Giancola and Bernard Ghanem. Temporally-aware feature pooling for action spotting in soccer broadcasts. *arXiv preprint arXiv:2104.06779*, 2021.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[10] Samuel Hurault, Coloma Ballester, and Gloria Haro. Self-supervised small soccer player detection and tracking. In *Proceedings of the 3rd International Workshop on Multimedia Content Analysis in Sports*, pages 9–18, 2020.

[11] Dippal Israni and Hiren Mewada. Feature descriptor based identity retention and tracking of players under intense occlusion in soccer videos. *International Journal of Intelligent Engineering and Systems*, 11(4):31–41, 2018.

[12] Wei Jiang, Juan Camilo Gamboa Higuera, Baptiste Angles, Weiwei Sun, Mehrsan Javan, and Kwang Moo Yi. Optimizing through learned errors for accurate sports field registration. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 201–210, 2020.

[13] Yudong Jiang, Kaixu Cui, Leilei Chen, Canjin Wang, and Changliang Xu. Soccerdb: A large-scale database for comprehensive video understanding. In *Proceedings of the 3rd International Workshop on Multimedia Content Analysis in Sports*, MMSports '20, page 1–8, New York, NY, USA, 2020. Association for Computing Machinery.

[14] Jacek Komorowski, Grzegorz Kurzejamski, and Grzegorz Sarwas. Footandball: Integrated player and ball detector. *arXiv preprint arXiv:1912.05445*, 2019.

[15] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*. Springer International Publishing, 2014.

[17] Keyu Lu, Jianhui Chen, James J Little, and Hangen He. Light cascaded convolutional neural networks for accurate player detection. *arXiv preprint arXiv:1709.10230*, 2017.

[18] M Manafifard, Hamid Ebadi, and H Abrishami Moghaddam. Multi-player detection in soccer broadcast videos using a blob-guided particle swarm optimization method. *Multimedia Tools and Applications*, 76(10):12251–12280, 2017.

[19] Pier Luigi Mazzeo, Paolo Spagnolo, Marco Leo, and Tiziana D'Orazio. Visual players detection and tracking in soccer matches. In *2008 IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance*, pages 326–333. IEEE, 2008.

[20] Kate McSurley and Greg Rybarczyk. An introduction to fieldf/x. 2011.

[21] K. Osumi, T. Yamashita, and H. Fujiyoshi. Domain adaptation using a gradient reversal layer with instance weighting. In *2019 16th International Conference on Machine Vision Applications (MVA)*, pages 1–5, 2019.

[22] Shafin Rahman, Salman Khan, and Nick Barnes. Polarity loss for zero-shot object detection. *arXiv preprint arXiv:1811.08982*, 2018.

[23] Shafin Rahman, Salman Khan, and Nick Barnes. Transductive learning for zero-shot object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6082–6091, 2019.

[24] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[26] Douglas A Reynolds. Gaussian mixture models. *Encyclopedia of biometrics*, 741:659–663, 2009.

[27] Aruni RoyChowdhury, Prithvijit Chakrabarty, Ashish Singh, SouYoung Jin, Huaizu Jiang, Liangliang Cao, and Erik Learned-Miller. Automatic adaptation of object detectors to new domains using self-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 780–790, 2019.

[28] Boris Sekachev, Nikita Manovich, Maxim Zhiltsov, Andrey Zhavoronkov, Dmitry Kalinin, Ben Hoff, TOsmanov, Dmitry Kruchinin, Artyom Zankevich, DmitriySidnev, Maksim Markelov, Johannes222, Mathis Chenuet, a andre, telenachos, Aleksandr Melnikov, Jijoong Kim, Liron Ilouz, Nikita Glazov, Priya4607, Rush Tehrani, Seungwon Jeong, Vladimir Skubriev, Sebastian Yonekura, vugia truong,

zliang7, lizhming, and Tritin Truong. opencv/cvat: v1.1.0, Aug. 2020.

[29] StatsPerform. Sports data - sports ai, technology, data feeds. Accessed: 2021-04-07.

[30] Nate Veldt, David F. Gleich, and Anthony Wirth. A correlation clustering framework for community detection. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 439–448, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

[31] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017.

[32] Ying Yang and Danyang Li. Robust player detection and tracking in broadcast soccer video based on enhanced particle filter. *Journal of Visual Communication and Image Representation*, 46:81–94, 2017.

[33] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[34] Liang Zheng, Zhi Bie, Yifan Sun, Jingdong Wang, Chi Su, Shengjin Wang, and Qi Tian. Mars: A video benchmark for large-scale person re-identification. In *European Conference on Computer Vision*, pages 868–884. Springer, 2016.