Multimodal and multiview distillation for real-time player detection on a football field

Anthony Cioppa*
University of Liège
anthony.cioppa@uliege.be

Adrien Deliège*
University of Liège
adrien.deliege@uliege.be

Noor Ul Huda
Aalborg University
nuh@create.aau.dk

Rikke Gade
Aalborg University

Marc Van Droogenbroeck
University of Liège

Thomas B. Moeslund
Aalborg University

Abstract

Monitoring the occupancy of public sports facilities is essential to assess their use and to motivate their construction in new places. In the case of a football field, the area to cover is large, thus several regular cameras should be used, which makes the setup expensive and complex. As an alternative, we developed a system that detects players from a unique cheap and wide-angle fisheye camera assisted by a single narrow-angle thermal camera. In this work, we train a network in a knowledge distillation approach in which the student and the teacher have different modalities and a different view of the same scene. In particular, we design a custom data augmentation combined with a motion detection algorithm to handle the training in the region of the fisheye camera not covered by the thermal one. We show that our solution is effective in detecting players on the whole field filmed by the fisheye camera. We evaluate it quantitatively and qualitatively in the case of an online distillation, where the student detects players in real time while being continuously adapted to the latest video conditions.

1. Introduction

Local sports fields can be expensive to construct and maintain, especially those built with artificial turf. Therefore, it is important to monitor and then optimize the occupancy of existing fields and stadiums. Furthermore, an automatic occupancy analysis method may open up new possibilities within real-time information and booking. In this work we propose a robust and cost-effective method for player detection and counting in a football field.

(*) Denotes equal contributions. Code at https://github.com/cioppaanthony/multimodal-multiview-distillation.

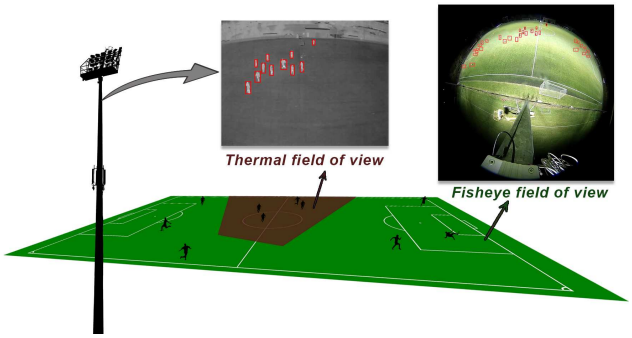

Figure 1: Illustration of the problem handled in this paper. We leverage the detections made on a thermal image on a part of the field to detect all the players on the whole field on the fisheye image.

For robust video monitoring of outdoor football fields, one main challenge is the size of the field. A field may be covered by either several regular cameras, which makes the setup rather complex and expensive, or it is possible to use a camera with a wide field of view, such as a fisheye camera. However, with a fisheye camera covering the entire football field, the players will appear small and have different orientation in the image due to the lens distortion. Player detection on these types of images is therefore not a trivial task. Another main challenge in outdoor environments is varying lighting conditions. Even though a football field may be illuminated during nights, lighting conditions will change during the day due to changing weather, position of the sun, and the effect of artificial lighting. To avoid problems with difficult lighting conditions, thermal cameras may be considered. These cameras capture only thermal infrared radiation, which represents temperature in the scene, hence they are more independent of lighting and normally eases the task of person detection because people have a

temperature different from the background [14]. However, thermal cameras are expensive and due to their limited field of view and resolution, several cameras would be needed to cover a football field.

To construct a camera setup that is reasonable in price level and at the same time robust to changes in weather and lighting conditions, we propose to use one fisheye RGB and one thermal camera co-located at the side of the field. An illustration of the setup and example images from the two cameras are shown in Figure 1. Only the fisheye camera will cover the entire field, while the detections obtained directly from the thermal camera will serve to provide some kind of ground truth for teaching a network.

There are two main contributions in this paper: (i) We show how two different image modalities and fields of view can be combined in a student-teacher distillation approach. (ii) We show how a student network can be trained to detect players outside the field of view of the teacher, through a combination of a custom data augmentation process and a motion detection algorithm.

2. Related work

Player detection in sports. Detection of players in sports fields is the first step of vision systems for sports applications, like occupancy analysis, tracking, performance analysis, etc. [36]. Background subtraction based methods have often been used for player detection due to the fast processing time that makes it well-suited for real-time applications. It has been applied for static cameras [1, 33] and for moving cameras in the case of uniformly colored surfaces [31]. However, noise should be expected due to, *e.g.*, other moving objects, similar colors in foreground and background, changing lighting conditions, and shadows. It has also been proposed to use classic person detection methods like using the AdaBoost algorithm for training a linear classifier with HOG features for detecting players in Australian Rules Football [11], or similarly with AdaBoost and Haar features for player detection in basketball [21] and baseball [26].

More recently, like for general object detection, CNN-based methods have also been the dominant trend for detecting sports players. In [34] a shallow CNN was trained to detect players on a hockey field, while others use pre-trained networks like Mask R-CNN for handball videos [30] and basketball videos [41], or YOLO for handball videos [6]. In [43] a reverse connected convolutional neural network (RC-CNN) is proposed for player detection. The reverse connected modules are embedded into the CNN to pass semantic information captured by deep layers back to shallower layers.

Person detection in fisheye and thermal cameras. Fisheye cameras have been widely used for person detection because of their advantage of wide viewing angle. Methods using a single camera setup have been reported for surveillance [22, 23], automobiles [24], indoor environment [35, 39] and outdoor sports field [19]. In these methods, the setup was used for pedestrian detection, tracking and occupancy analysis. Multiple camera setups are also proposed to detect persons for similar applications [3, 28, 40]. However, the main disadvantages with fisheye cameras are the distortion on the borders and the lower image quality in low lighting conditions.

Thermal cameras have long been used in practice because of their efficiency in bad lighting conditions. The range of applications varies from industrial uses to daily life traffic and surveillance [14]. Various methods based on thermal cameras have been proposed for person detection, such as feature extraction and threshold based methods [9, 12, 13, 42], HOG methods [25, 37], machine learning techniques [18] and deep neural networks [16, 17, 20]. A dataset and a trained network for people detection on outdoor thermal images have been proposed in [20]. The disadvantage of thermal cameras is their expensive cost and their reduced field of view.

In this work we will continue on recent trends to use a CNN-based method for player detection. We aim to circumvent the limitations of both fisheye and thermal cameras, by combining these modalities and teach the network for the fisheye camera with detections from the thermal camera, in a student-teacher distillation approach.

3. Data acquisition and calibration

Camera setup. The data used in this work consist of video streams of two different cameras: a fisheye camera and a thermal camera. Both cameras are installed on the same pole at the side of a football field, as illustrated in Figure 1. The thermal camera is placed approximately 9.8 meters above the ground and the fisheye camera is installed at 9.5 meters. By doing so, the field of view of the fisheye camera covers the whole football field, whereas the thermal camera covers the central area, as shown in Figure 1. In this setup, the field of view of the thermal camera represents 6% of the fisheye image, and covers 22% of the football field as seen by the fisheye camera. Let us note that several teams use the field simultaneously for a training session during the video. Hence, the players are performing different activities, such as moving goals or performing various exercises. Therefore, the players can be found in different postures in any part of the field.

Acquisition. The fisheye video stream is recorded using a Hikvision Fisheye Network Camera with a resolution of 1280×1280 pixels and a field of view of $360°$. The thermal video stream is recorded using an Axis Q1922 camera that has a resolution of 640×480 pixels and $57°$ of horizontal viewing angle. The videos were recorded during one

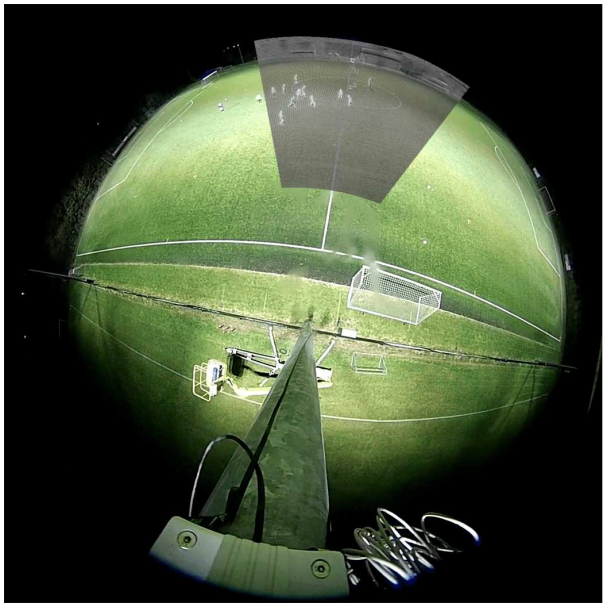

Figure 2: Projection of the thermal image onto the fisheye image. The thermal camera sees only $\approx 22\%$ of the football field pixels of the fisheye image.

hour in an amateur football field in December 2017, at night time with artificial light illuminating the field. The fisheye camera records the video at 12 fps. The thermal camera initially records the video at 30 fps, which is then re-sampled at 12 fps to allow a synchronization of the two streams. A proper camera calibration and registration between fisheye and thermal images is required for the transferability of points of interest.

Calibration and registration. First, a calibration of the internal parameters of each camera is performed following the procedure described in [29]. For the thermal camera, an A3-sized 10 mm polystyrene foam board is used as backdrop and a board of the same size with cut-out squares is used as checkerboard. In order to obtain a suitable contrast, the backdrop is heated and the checkerboard is placed at room temperature before the calibration. For the fisheye camera calibration, a checkerboard of 25×25 centimeters is used. Finally, the camera parameters derived from the calibration are obtained with a Matlab toolbox [4].

Second, we perform the registration between the two cameras. We undistord the images of the cameras using the internal parameters obtained previously. We manually choose several points of interest on the undistorded football field to compute the homography between the cameras, following [27]. These points are player feet positions for the players seen by the two cameras. The projection of the thermal image onto the fisheye image is shown in Figure 2.

4. Methodology

Problem statement. A general formulation of the problem tackled in this paper is the following. Given a network performing a detection task on data from a camera, how can we train a real-time network for the same detection task on data from another camera with a possibly different modality and a different field of view of the same scene? In this section, we describe our solution for this problem in general terms, and we also explain how each step is particularized for our practical use case. Our use case consists in the task of player detection on a football field given a network able to detect players on a fixed thermal camera with a narrow field of view, which is used to train another detection network on data from a fixed fisheye camera with a wide field of view. This is illustrated in Figure 1.

Notations. We handle this problem with a teacher-student distillation approach, in which the output of a trained teacher network $\mathcal{T}$ serves as surrogate ground truth to train a student network $\mathcal{S}$ (see [38] for a recent review). Such a method has already been successfully applied in sports in [7] for segmenting football and basketball players in real time by distilling a slow $\mathcal{T}$ (Mask R-CNN [15]) into a fast $\mathcal{S}$ (TinyNet [8]). In addition, in [7], the distillation is performed in an online fashion, such that $\mathcal{S}$ continuously adapts to the latest game conditions. However, $\mathcal{T}$ and $\mathcal{S}$ use the same video feed, which implies that $\mathcal{S}$ can be directly (no transformation needed) and entirely (no missing ground truth) supervised by $\mathcal{T}$.

In the present work, the setup is more challenging as $\mathcal{T}$ and $\mathcal{S}$ process the video feeds of two cameras $\mathcal{C}_\mathcal{T}$ and $\mathcal{C}_\mathcal{S}$ with different modalities and fields of view. Having different modalities prevents us from using $\mathcal{T}$ on the feed of $\mathcal{C}_\mathcal{S}$, and having different fields of view prevents us from directly and entirely supervising $\mathcal{S}$. We assume that $\mathcal{C}_\mathcal{T}$ and $\mathcal{C}_\mathcal{S}$ are synchronized, such that they capture frames $\mathcal{C}_\mathcal{T}(t)$ and $\mathcal{C}_\mathcal{S}(t)$ simultaneously at each capture time t. We also assume that the projection from $\mathcal{C}_\mathcal{T}(t)$ to $\mathcal{C}_\mathcal{S}(t)$, expressed in terms of pixel coordinates, is known from the preliminary calibration step explained in the previous section. We note $\mathbb{P}$ the area of $\mathcal{C}_\mathcal{S}(t)$ representing the projection on $\mathcal{C}_\mathcal{S}(t)$ of the part of the scene also filmed by $\mathcal{C}_\mathcal{T}$ (shown in Figure 3). The remaining part of $\mathcal{C}_\mathcal{S}(t)$ is filmed by $\mathcal{C}_\mathcal{S}$ only and is noted $\overline{\mathbb{P}}$. As both cameras are fixed, this partition of $\mathcal{C}_\mathcal{S}(t)$ is independent of t.

In order to train $\mathcal{S}$, we need surrogate ground-truth bounding boxes both in $\mathbb{P}$ and in $\overline{\mathbb{P}}$. We detail hereafter how we obtain such boxes in $\mathcal{C}_\mathcal{S}(t)$ for a given capture time t. Following common practice, we represent a bounding box coordinates by a quadruplet containing the two coordinates of the center of the box, its width and its height.

Surrogate ground truths in $\mathbb{P}$. This part is straightforward. First, we use $\mathcal{T}$ to detect players in $\mathcal{C}_\mathcal{T}(t)$ and retrieve the

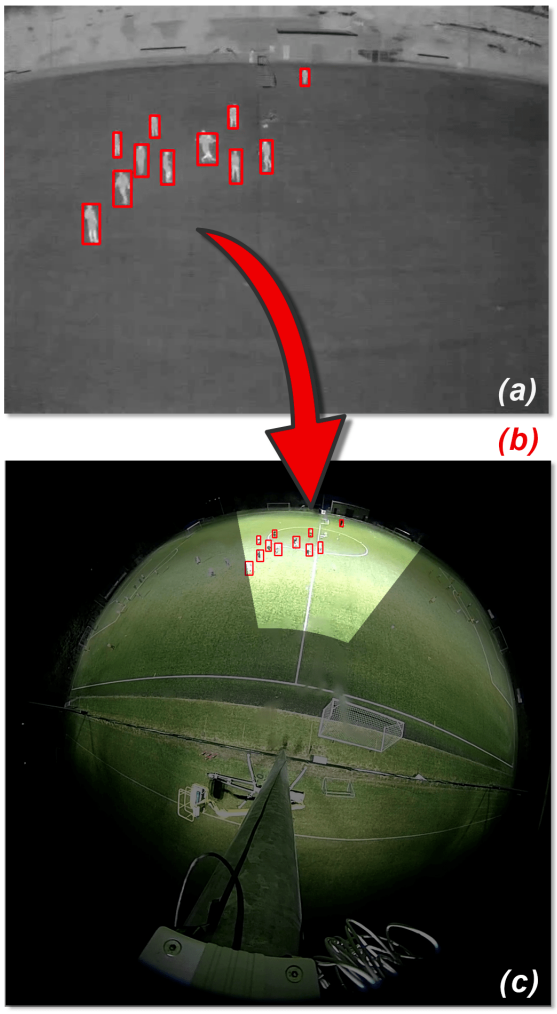

Figure 3: The bounding boxes given by $\mathcal{T}$ on $\mathcal{C}_{\mathcal{T}}(t)$ (a) are projected (b) into $\mathcal{C}_{\mathcal{S}}(t)$ to provide us surrogate ground-truth bounding boxes in $\mathbb{P}$ (c).

coordinates of bounding boxes of $\mathcal{C}_{\mathcal{T}}(t)$. Then, we project them into $\mathcal{C}_{\mathcal{S}}(t)$ using the calibration of the previous section. By doing so, we obtain the surrogate ground-truth bounding boxes of $\mathcal{C}_{\mathcal{S}}(t)$ that are located in $\mathbb{P}$, as shown in Figure 3. The remaining part of $\mathbb{P}$ constitutes detection-free areas.

Surrogate ground truths in $\overline{\mathbb{P}}$. This part is more difficult as we cannot have a direct access to the pixels of $\overline{\mathbb{P}}$ from those of $\mathcal{C}_{\mathcal{T}}(t)$. Training $\mathcal{S}$ solely with the boxes provided in $\mathbb{P}$ for each $\mathcal{C}_{\mathcal{S}}(t)$ leads the network to focus only on $\mathbb{P}$ and to overlook $\overline{\mathbb{P}}$ for each frame. Eventually, the network is not able to detect anything in $\overline{\mathbb{P}}$.

To circumvent this problem, our idea is the following. First, we use a custom data augmentation process to create artificial players with known bounding boxes in $\overline{\mathbb{P}}$. This provides us the "ground-truth locations" of some "true positive" players that $\mathcal{S}$ will have to detect. This is not sufficient as we still need "ground-truth information" in areas where we did not create any player. For that purpose, we use a motion detection algorithm to identify areas of $\overline{\mathbb{P}}$ that are guaranteed player-free. This provides us "true negative" areas, in which $\mathcal{S}$ will be penalized when predicting player bounding boxes. In the remaining areas of $\overline{\mathbb{P}}$, we have no useful information, hence $\mathcal{S}$ will not be penalized. These two steps are described in detail hereafter.

[1. Custom data augmentation] In order to introduce true positive players with known bounding boxes in $\overline{\mathbb{P}}$, we design the following automatic data augmentation process. Given a frame $\mathcal{C}_{\mathcal{S}}(t)$, we start by randomly extracting image crops delimited either by one isolated or by several adjacent bounding boxes previously obtained in $\mathbb{P}$ (Figure 4). Then, for each crop, we randomly select a pixel in $\overline{\mathbb{P}}$, which will serve as an anchor point where the crop will be pasted after being rescaled and rotated appropriately. In our use case, the anchors are selected in the subset of $\overline{\mathbb{P}}$ corresponding to the football field.

We perform a rescaling and a rotation of each crop to produce an insertion that looks as realistic as possible by taking into account the inherent distortions of $\mathcal{C}_{\mathcal{S}}$ (Figure 4). Let (r, θ) denote the initial polar coordinates (with origin located at the center of $\mathcal{C}_{\mathcal{S}}(t)$) of the center of the crop and (r', θ') those of its selected anchor point. We rescale the crop by a factor $\alpha e^{\beta(r'-r)} + \gamma$ with $\alpha = 0.5, \beta = -0.004, \gamma = 0.5$ and rotate it by the angle difference $\theta' - \theta$. Finally, we paste the transformed crop on $\mathcal{C}_{\mathcal{S}}(t)$ itself with OpenCV's seamless blending function, such that its center is located at the selected anchor point (Figure 4). In order to obtain the boxes associated with these artificial players, we perform the same transformation on each bounding box included in the initial crop. Eventually, for each transformed box, we consider as surrogate ground-truth bounding box the smallest unrotated (regular) rectangular box that encloses it (Figure 4).

In our fisheye setup, the data augmentation process allows to create artificial players with known bounding boxes in $\overline{\mathbb{P}}$ (Figure 4). However, this does not suffice to train $\mathcal{S}$ efficiently, as real players without known boxes may still be present in $\overline{\mathbb{P}}$. In a standard training process, $\mathcal{S}$ would thus be forced to detect the artificial players and would be penalized for detecting the remaining real ones. To bypass this undesirable effect, we remove the penalty suffered by $\mathcal{S}$ for detections containing enough motion. Hence, we leverage a motion detection algorithm to determine where this should be applied. By doing so, we also obtain areas where there is assuredly no player, *i.e.* where detections should not be made.

[2. Motion detection] As we handle a video feed from a fixed camera, we use ViBe [2] to obtain, for each frame

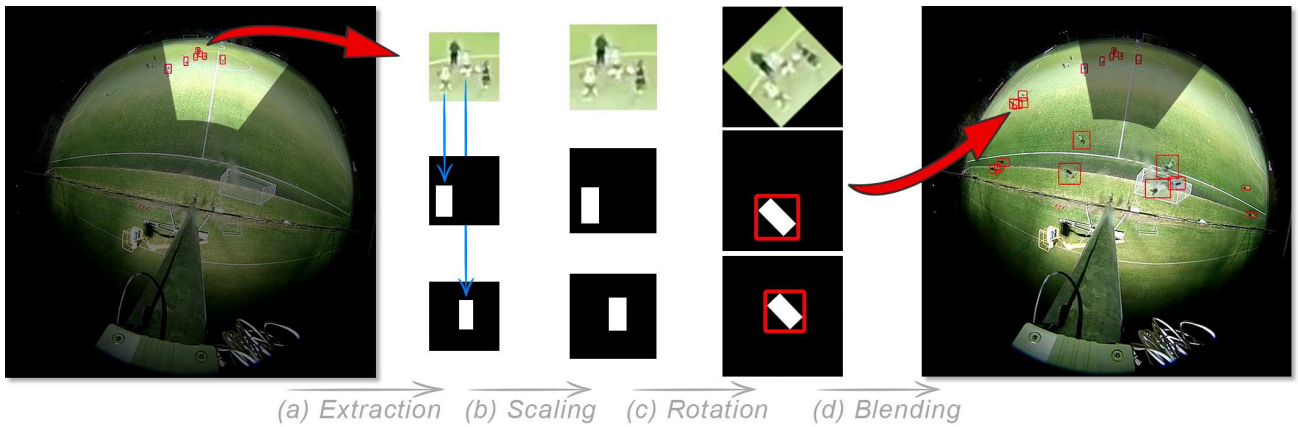

Figure 4: Our custom data augmentation pipeline designed to construct surrogate ground-truth bounding boxes in the region $\overline{\mathbb{P}}$ filmed by $\mathcal{C}_\mathcal{S}$ only. First, crops containing players are extracted (a) from the area filmed by both cameras $\mathbb{P}$, in which we know their location. Then, each crop and its associated bounding boxes are scaled (b) and rotated (c) to be appropriately pasted in $\overline{\mathbb{P}}$. A seamless blending is applied during the collage to increase the realistic aspect of the augmented image. As a result, we create artificial players with known bounding boxes in $\overline{\mathbb{P}}$.

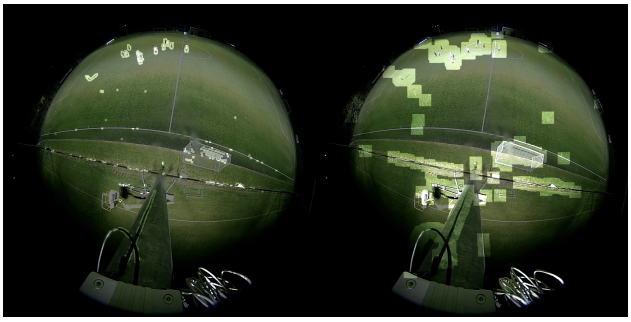

Figure 5: Initial motion detection mask $\mathrm{M}(t)$ overlayed on its corresponding frame (left), and enlarged motion detection mask $\mathbb{M}(t)$ (right).

$\mathcal{C}_\mathcal{S}(t)$, the set of pixels that are in motion, noted $\mathrm{M}(t)$, and those that are not, noted $\overline{\mathrm{M}(t)}$ (Figure 5). ViBe is very sensitive to motion, which implies that, in our fisheye setup, $\mathrm{M}(t)$ almost surely contains all the players, as well as pixels corresponding to the balls, player shadows, and some noise. As $\mathrm{M}(t)$ may be tight around the players, we morphologically dilate it by a 11×11 square kernel to ensure that it includes the bounding boxes that would surround the players if they were available (Figure 5). By doing so, we obtain an enlarged mask $\mathbb{M}(t)$, such that we can penalize $\mathcal{S}$ when it detects players in $\overline{\mathbb{M}(t)}$, *i.e.* outside the enlarged mask. However, $\mathbb{M}(t)$ remains an area of uncertainty, where we do not penalize $\mathcal{S}$. Technically, this means that we zero out the loss in this area during training, as detailed hereafter.

Training $\mathcal{S}$. We use the YOLOv3 network [32] trained to detect humans on thermal images in [20] as teacher network $\mathcal{T}$. We use YOLOv3-tiny [32] as student network $\mathcal{S}$, adapted for a single class problem and with four times less channels for each convolutional layer. Hence, $\mathcal{S}$ outputs a list of 5-dimensional vectors. Each of them encapsulates information on a predicted bounding box: the four coordinates (x, y, w, h) defining the box, and a player score p representing its confidence for a player to actually belong to the box.

The loss of YOLOv3-tiny, hence $\mathcal{S}$, penalizes these vectors in the following way (see Figure 6). For a predicted box close to a surrogate ground-truth box (either in $\mathbb{P}$ or in $\overline{\mathbb{P}}$), the mean square error loss between the coordinates of the boxes is computed, as well as the binary cross-entropy loss of p. This encourages the network to predict a high confidence score (closer to 1) and to find the right dimensions of the box. For a box far from a surrogate ground-truth box, only the binary cross-entropy loss of $1 - p$ is computed, to discourage the network from predicting a player in that box (p closer to 0). In our case, we must take into account the uncertainty about the boxes in $\mathbb{M}(t)$ in the region $\overline{\mathbb{P}}$, as they may correspond to unnanotated real players. Therefore, for a box far from a surrogate ground-truth box (including those created by the data augmentation), we zero out its loss if the center of the box is in $\overline{\mathbb{P}}$ and is in motion (belongs to $\mathbb{M}(t)$). If the center of the box belongs to $\overline{\mathbb{M}(t)}$, we are practically sure that there is no player in the box, and we thus leave the loss as is to penalize that detection. There is not particular restriction about the loss in $\mathbb{P}$. This is illustrated in Figure 6.

Inference. When used for inference, we verify that the bounding boxes predicted by $\mathcal{S}$ contain enough motion. Indeed, the predicted boxes whose center is not in motion, *i.e.* outside $\mathbb{M}(t)$, are not likely to contain a player. Therefore they are removed from the final output of $\mathcal{S}$.

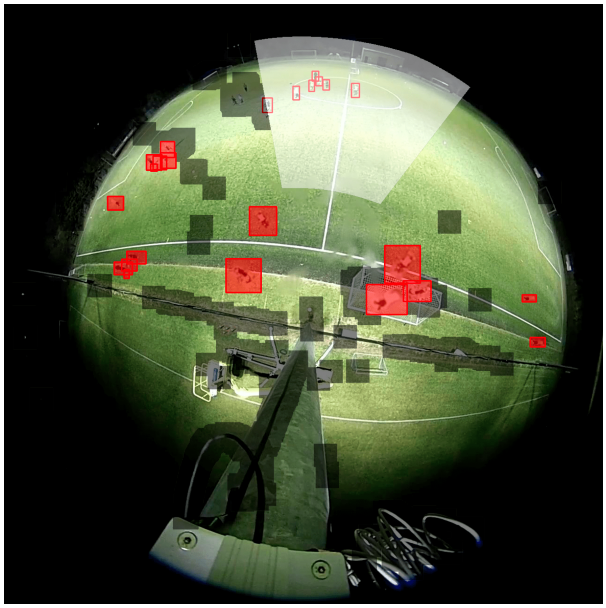

Figure 6: Combination of our data augmentation and motion detection algorithms, showing how the loss is applied to penalize the predictions of $\mathcal{S}$ in $\overline{\mathbb{P}}$ (outside the white area). $\mathcal{S}$ must detect the players artificially created (red rectangles). Also, predicted boxes whose center falls within the enlarged motion mask $\mathbb{M}(t)$ (the black zones) do not generate any loss, since this area includes the players of $\overline{\mathbb{P}}$ not erased by the data augmentation, for which we have no ground-truth boxes. Finally, $\mathcal{S}$ must not predict any box in the rest of the image in $\overline{\mathbb{P}}$. Let us recall that the loss is applied everywhere in $\mathbb{P}$, as we have the ground truth from $\mathcal{T}$ in that area.

5. Experiments

Online distillation. In this work, we perform the distillation of the teacher network $\mathcal{T}$ into the student network $\mathcal{S}$ in an online manner as in [7]. The reason for using that process is threefold. First, this allows $\mathcal{S}$ to continuously adapt to the latest weather and lighting conditions. Second, in a real-life deployment of the system, the online distillation will indeed be performed continuously. Hence, in order to have an understanding of how $\mathcal{S}$ behaves as it trains and detects people in real time, it is worth testing $\mathcal{S}$ under similar conditions. Third, training $\mathcal{S}$ adaptively allows us to study the evolution of the performance of the network as it learns through time. As we have only one video sequence with both the thermal and the fisheye recordings, this also enables us to evaluate $\mathcal{S}$ multiple times rather than measuring its performance only once, on a unique (and maybe abnormally hard or easy) small set of frames.

In the online distillation process, all the frames of the fisheye camera $\mathcal{C}_\mathcal{S}$ are treated by $\mathcal{S}$, which runs in real time. Meanwhile, some frames of the video feed of the thermal camera $\mathcal{C}_\mathcal{T}$ are input to $\mathcal{T}$, which provides boxes converted into surrogate ground-truth bounding boxes in the area $\mathbb{P}$ of the frame captured by $\mathcal{C}_\mathcal{S}$. These boxes are accumulated in an online dataset with 5-minutes memory, and the dataset is used to train a copy of $\mathcal{S}$ in a separate thread. The training is performed on the whole frames $\mathcal{C}_\mathcal{S}(t)$ as described in the previous section, using our data augmentation and motion detection processes outside $\mathbb{P}$. When this copy of $\mathcal{S}$ has trained during one epoch on the online dataset, its weights are updated and transferred into the initial network $\mathcal{S}$ that performs the detection on all the frames. Consequently, the weights of this network evolves through time to continuously adapt to the latest video conditions.

Quantitative evaluation. To assess the performance of the student network $\mathcal{S}$ over the course of the video, we manually annotated the ground-truth bounding boxes for all the players of one frame every 10 seconds of the fisheye video. We compute the performance of $\mathcal{S}$ on a set of frames with the Average Precision (AP) metric particularized for one class. Following practice for the Pascal VOC dataset [10], each bounding box predicted by $\mathcal{S}$ is matched with the ground-truth box with which it has the largest intersection over union (IoU). We consider predicted boxes with an IoU larger than some threshold t_IoU as true positives, the others as false positives, and the ground-truth boxes left unmatched are false negatives. If several true positives are associated with the same ground-truth box, only one of them is kept as a true positive, while the others are rather considered as false positives. We note the number of true positives (resp. false positives, false negatives) TP (resp. FP, FN). Then, we compute the precision and recall as

$$P = \frac{TP}{TP + FP} \quad \text{and} \quad R = \frac{TP}{TP + FN}.$$

We compute the points (P, R) for various thresholds on the confidence scores of the boxes to obtain the PR curve. Finally, we compute the area under the PR curve as suggested in [10] to obtain the AP for that set of frames. Despite its limitations [5], this kind of evaluation process has been widely adopted in the community.

In order to determine an appropriate value of t_IoU for evaluating the performance of $\mathcal{S}$, we examine the efficiency of $\mathcal{T}$ in predicting the boxes in $\mathbb{P}$. For that purpose, we compute the AP of $\mathcal{T}$ on the last 15 minutes of video, for several values of t_IoU ranging from 0 to 1, for the frames where ground-truth annotations are available. This allows us to determine how good $\mathcal{T}$ is at centering its bounding boxes on the players. The performance of $\mathcal{T}$ in $\mathbb{P}$ as a function of t_IoU is shown in Figure 7. We can see that $\mathcal{T}$ is not perfect in $\mathbb{P}$, which conditions the performances that can be expected from $\mathcal{S}$. To evaluate $\mathcal{S}$, we choose t_IoU $= 0.25$, as $\mathcal{T}$ displays reasonable performances in $\mathbb{P}$ with that threshold. Given the small size of the boxes, it also makes sense

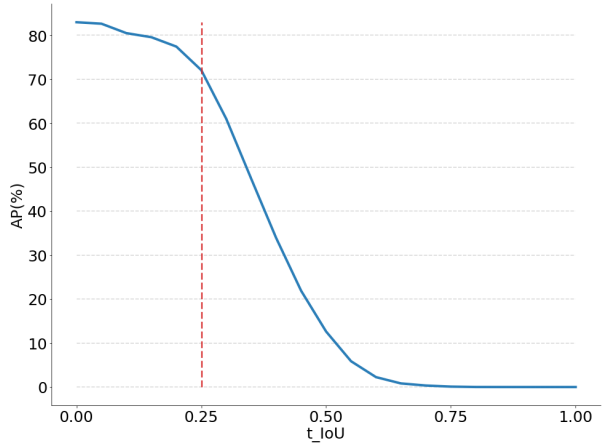

Figure 7: **Performances of $\mathcal{T}$ in $\mathbb{P}$** on the last 15 minutes of video as a function of t_IoU. This quantifies how accurately $\mathcal{T}$ centers its bounding boxes on the players. We can see that $\mathcal{T}$ is not perfect. We decide to evaluate the **performances of $\mathcal{S}$ for t_IoU** $= 0.25$, as we consider it as the largest t_IoU for which $\mathcal{T}$ still displays satisfying performances (AP $> 70\%$).

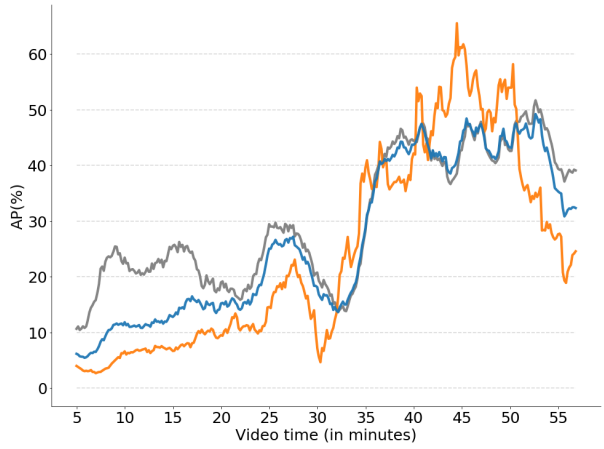

Figure 8: Evolution of the performances of the student network $\mathcal{S}$ through the video in $\mathbb{P}$, $\overline{\mathbb{P}}$, and in the **whole frames**. We can see that the network improves over time and that it manages to perform well both in $\mathbb{P}$ and in $\overline{\mathbb{P}}$.

to examine the performance of $\mathcal{S}$ for a relatively low value of t_IoU. Let us recall that the boxes outputted by the network are independent of any particular choice of threshold. It serves only for quantitative evaluation purposes.

Following [7], we evaluate the performance of the student network $\mathcal{S}$ progressively. Every 10 seconds, $\mathcal{S}$ predicts the bounding boxes of the frames for which we have manual annotations within a running temporal window that covers the next 3 minutes of video. For this set of frames, we compute the AP. The evolution on the AP through time with

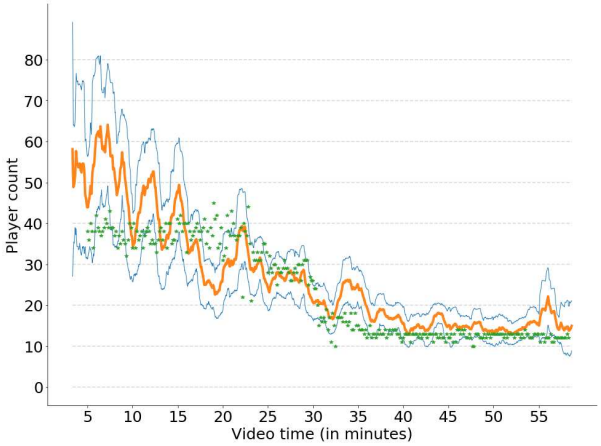

Figure 9: **Results on the player counting task** averaged over a 1-minute window, and associated **standard deviation**. During the last 15 minutes, we have a RMSE with the **ground truth** of 3.4 players, which is reasonable and shows that our method provides a reliable estimate of the occupancy of the football field.

t_IoU $= 0.25$ is represented in Figure 8. We see that the performance tends to increase, indicating that $\mathcal{S}$ learns to better detect players over time. Figure 8 also reveals that there is still room for improvement in the present challenge.

We further examine the effectiveness of our data augmentation and motion detection processes to train $\mathcal{S}$ for detecting players outside $\mathbb{P}$. For that purpose, we perform a region-specific analysis by computing the temporal evaluation of the AP within $\mathbb{P}$ and $\overline{\mathbb{P}}$. The performance curves are displayed in Figure 8. We note that $\mathcal{S}$ learns efficiently to detect players in $\overline{\mathbb{P}}$, as the performances for $\mathbb{P}$ and $\overline{\mathbb{P}}$ are close to each other and follow the same trend. Also, further experiments reveal that the post-processing with the motion mask $\mathbb{M}(t)$ is particularly helpful to increase the performance in $\overline{\mathbb{P}}$. In that area, the AP decreases by 5 to 20% without post-processing, while the drop is below 3% in $\mathbb{P}$.

Finally, as a potential application of this system is to monitor the use of the football field, we examine the results obtained for the task of people counting. The predicted number of people on the field corresponds to the number of bounding boxes predicted by $\mathcal{S}$ (thus on the fisheye images) after post-processing. We average the counting using a 1-minute sliding window. The results are displayed in Figure 9. We note that our method gives a globally reliable estimate of the number of people present on the field. Quantitatively, during the last 15 minutes of video, the root mean square error (RMSE) between the predictions and the ground truth is as low as 3.4 players. Again, we can see that the performance tends to increase over time since the estimate is more accurate at the end of the video, indicating that $\mathcal{S}$ learns to better detect players over time. Also,

In $\overline{\mathbb{P}}$	With data augmentation	Without data augmentation
Cancel loss in the motion mask $\mathbb{M}(t)$	**Our full method.** Most players in $\overline{\mathbb{P}}$ correctly detected, few false positives.	Few players detected in $\overline{\mathbb{P}}$, unusable in practice
Activate loss everywhere in $\overline{\mathbb{P}}$	Able to detect players in $\overline{\mathbb{P}}$, but not as good as our full method	Unable to make any detection in $\overline{\mathbb{P}}$, no true positives
Cancel loss everywhere in $\overline{\mathbb{P}}$	Thousands of detections in $\overline{\mathbb{P}}$, mostly false positives	Thousands of detections in $\overline{\mathbb{P}}$, mostly false positives

Table 1: Ablation results in $\overline{\mathbb{P}}$. The combination of the data augmentation and the motion detection algorithm gives the best trade-off between true and false positive detections.

we can see in Figure 9 that the standard deviation of the box count computed for each sliding window decreases over time, which indicates that the network becomes more consistent as it trains. Even though $\mathcal{S}$ tends to slightly overestimate the actual number of players, we can see that it manages to provides a good overview of the use of the field.

Qualitative evaluation. To further assess the usefulness of our data augmentation and motion detection processes, we perform ablation studies on the components of our method. We investigate the combination of either enabling or disabling the data augmentation, with either zeroing out the loss in the motion mask $\mathbb{M}(t)$, or nowhere in $\overline{\mathbb{P}}$, or everywhere in $\overline{\mathbb{P}}$. The effects observed for these setups are reported in Table 1. In our experiments, we observe that the combination of the data augmentation and of zeroing out the loss in $\mathbb{M}(t)$, as detailed in this paper, leads to the best student network $\mathcal{S}$ at inference time. Activating the loss everywhere in $\overline{\mathbb{P}}$ at training time forces $\mathcal{S}$ to detect only the artificial players in $\overline{\mathbb{P}}$ and to avoid detecting the actual players of $\overline{\mathbb{P}}$ that have not been erased by the data augmentation. This may confuse $\mathcal{S}$, leading to a decrease in its ability to detect players in $\overline{\mathbb{P}}$ at inference time. We notice that canceling the loss everywhere in $\overline{\mathbb{P}}$ leads to thousands of predicted bounding boxes in $\overline{\mathbb{P}}$ at inference time. This makes sense since the network is not forced to detect or not players in $\overline{\mathbb{P}}$ in this case. Most of these predictions are false positives, and the system is useless in practice. As indicated in Table 1, we also note that removing the data augmentation always leads to mediocre networks, for similar reasons as those already explained. In particular, activating the loss everywhere in $\overline{\mathbb{P}}$ makes $\mathcal{S}$ unable to detect any single player in $\overline{\mathbb{P}}$. This results from the absence of ground-truth true positives (both artificial and real ones) in $\overline{\mathbb{P}}$.

Finally, examples of detections provided by $\mathcal{S}$ are given in Figure 10. We can see that players located in $\overline{\mathbb{P}}$ are detected as efficiently as those located in $\mathbb{P}$. This was made possible thanks to our data augmentation and motion detection algorithms in the distillation approach.

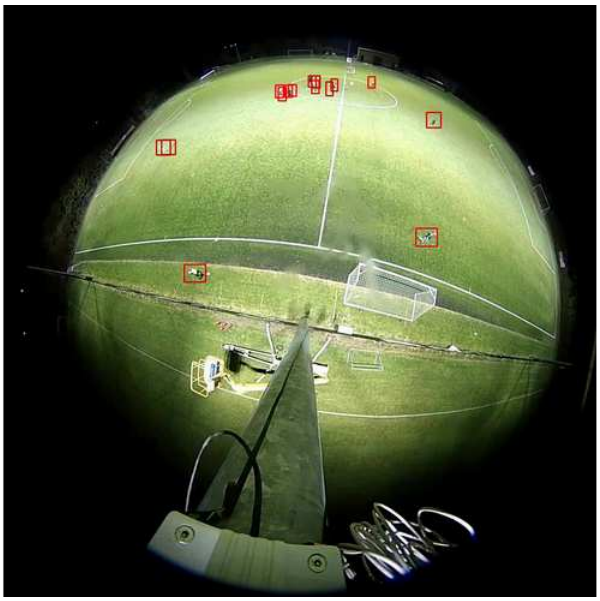

Figure 10: Detections on a test frame. We can note that players are accurately detected, even though there are a few superfluous predicted bounding boxes.

6. Conclusion

In this work, we propose a novel system for monitoring the field occupancy in low-budget football stadiums. Our system uses a single wide-angle fisheye camera assisted by a thermal camera to detect and count all the players on the field. We use a network trained in a student-teacher distillation approach. The student network is locally supervised by a teacher network that easily detects players on the thermal camera. These detections are then projected into the fisheye camera using camera registration and serve as surrogate ground truths. Since both cameras have different modalities and fields of view of the scene, the student cannot be fully supervised by the teacher. Therefore, we develop a custom data augmentation process, combined with motion information provided by a background subtraction algorithm, to introduce surrogate ground truths outside their common field of view. In our case, we perform the distillation in an online fashion, *i.e.* our student is continuously trained to adapt to the latest video conditions, while performing the player detection in real-time. We show that our system is able to accurately detect players both inside and outside the common field of view, thanks to our custom supervision.

Acknowledgments A. Cioppa is funded by the FRIA. A. Deliège is supported by the DeepSport project of the Walloon region, Belgium.

References

[1] M. Archana and M. Kalaiselvi Geetha. An efficient ball and player detection in broadcast tennis video. In *Intelligent Systems Technologies and Applications*, pages 427–436, Cham, 2016. Springer International Publishing. 2

[2] Olivier Barnich and Marc Van Droogenbroeck. ViBe: A universal background subtraction algorithm for video sequences. *IEEE Transactions on Image Processing*, 20(6):1709–1724, June 2011. 4

[3] Massimo Bertozzi, Luca Castangia, Stefano Cattani, Antonio Prioletti, and Pietro Versari. 360° detection and tracking algorithm of both pedestrian and vehicle using fisheye images. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 132–137, June 2015. 2

[4] Jean-Yves Bouguet. Camera calibration toolbox for Matlab, 2014. 3

[5] Kendrick Boyd, Vítor Santos Costa, Jesse Davis, and C. David Page. Unachievable region in precision-recall space and its effect on empirical evaluation. In *Proceedings of the 29th International Coference on International Conference on Machine Learning (ICML)*, ICML'12, page 1619–1626, Madison, WI, USA, 2012. Omnipress. 6

[6] Matija Buric, Marina Ivasic-Kos, and Miran Pobar. Player tracking in sports videos. In *IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pages 334–340, Dec. 2019. 2

[7] Anthony Cioppa, Adrien Deliège, Maxime Istasse, Christophe De Vleeschouwer, and Marc Van Droogenbroeck. ARTHuS: Adaptive Real-Time Human Segmentation in Sports Through Online Distillation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 3, 6, 7

[8] Anthony Cioppa, Adrien Deliège, and Marc Van Droogenbroeck. A bottom-up approach based on semantics for the interpretation of the main camera stream in soccer games. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1846–1855, June 2018. 3

[9] Congxia Dai, Yunfei Zheng, and Xin Li. Layered representation for pedestrian detection and tracking in infrared imagery. In *IEEE Conference on Computer Vision and Pattern Recognition(CVPR) Workshops*, Sep. 2005. 2

[10] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 6

[11] Hayden Faulkner and Anthony Dick. AFL player detection and tracking. In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, Nov. 2015. 2

[12] Rikke Gade, Anders Jørgensen, and Thomas B. Moeslund. Occupancy analysis of sports arenas using thermal imaging. In *International Conference on Computer Vision Theory and Applications*, pages 277–283. SCITEPRESS Digital Library, 2012. 2

[13] Rikke Gade, Anders Jørgensen, and Thomas B. Moeslund. Long-term occupancy analysis using graph-based optimisation in thermal imagery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3698–3705, US, 2013. IEEE Computer Society Press. 2

[14] Rikke Gade and Thomas B. Moeslund. Thermal cameras and applications: A survey. *Machine Vision and Applications*, 25(1):245–262, 2014. 2

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, Oct. 2017. 3

[16] Duyoung Heo, Eunju Lee, and Byoung Chul Ko. Pedestrian detection at night using deep neural networks and saliency maps. *Journal of Imaging Science and Technology*, 61:60403–1–60403–9(9), 2017. 2

[17] Christian Herrmann, Thomas Müller, Dieter Willersinn, and Jürgen Beyerer. Real-time person detection in low-resolution thermal infrared imagery with MSER and CNNs. In *SPIE Security + Defence*, volume 9987, 2016. 2

[18] Noor Ul Huda, Kasper Halkjær Jensen, Rikke Gade, and Thomas B. Moeslund. Estimating the number of soccer players using simulation-based occlusion handling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1937–1946, US, 2018. IEEE. 2

[19] Noor Ul Huda, Bolette D. Hansen, Rikke Gade, and Thomas B. Moeslund. Occupancy analysis of soccer fields using wide-angle lens. In *International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*, pages 354–359, Dec. 2017. 2

[20] Noor Ul Huda, Bolette D. Hansen, Rikke Gade, and Thomas B. Moeslund. The effect of diverse dataset for transfer learning in thermal person detection. *Sensors*, 20(7):1982, Apr 2020. 2, 5

[21] Zdravko Ivankovic, Branko Markoski, Miodrag Ivkovic, Dragica Radosav, and Predrag Pecev. Adaboost in basketball player identification. In *IEEE International Symposium on Computational Intelligence and Informatics (CINTI)*, pages 151–156, Nov. 2012. 2

[22] Hyungtae Kim, Eunjung Chae, Gwanghyun Jo, and Joonki Paik. Fisheye lens-based surveillance camera for wide field-of-view monitoring. In *IEEE International Conference on Consumer Electronics (ICCE)*, pages 505–506, 2015. 2

[23] Hyungtae Kim, Jaehoon Jung, and Joonki Paik. Fisheye lens camera based surveillance system for wide field of view monitoring. *Optik*, 127(14):5636–5646, 2016. 2

[24] Dan Levi and Shai Silberstein. Tracking and motion cues for rear-view pedestrian detection. In *IEEE International Conference on Intelligent Transportation Systems*, pages 664–671, Sep. 2015. 2

[25] Wei Li, Dequan Zheng, Tiejun Zhao, and Mengda Yang. An effective approach to pedestrian detection in thermal imagery. In *International Conference on Natural Computation*, pages 325–329, May 2012. 2

[26] Zahid Mahmood, Tauseef Ali, and Shahid Khattak. Automatic player detection and recognition in images using adaboost. In *International Bhurban Conference on Applied Sciences Technology (IBCAST)*, pages 64–69, Jan. 2012. 2

[27] Ezio Malis and Manuel Vargas. Deeper understanding of the homography decomposition for vision-based control. Research Report RR-6303, INRIA, 2007. 3

[28] Van Tuan Nguyen, Thanh Binh Nguyen, and Sun-Tae Chung. ConvNets and AGMM based real-time human detection under fisheye camera for embedded surveillance. In *International Conference on Information and Communication Technology Convergence (ICTC)*, pages 840–845. IEEE, 2016. 2

[29] Cristina Palmero, Albert Clapés, Chris Bahnsen, Andreas Møgelmose, Thomas B. Moeslund, and Sergio Escalera. Multi-modal RGB-depth-thermal human body segmentation. *International Journal of Computer Vision*, 118(2):217–239, 2016. 3

[30] Miran Pobar and Marina Ivasic-Kos. Mask R-CNN and optical flow based method for detection and marking of handball actions. In *International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–6, Oct. 2018. 2

[31] Upendra M. Rao and Umesh C. Pati. A novel algorithm for detection of soccer ball and player. In *International Conference on Communications and Signal Processing (ICCSP)*, pages 344–348, Apr. 2015. 2

[32] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *CoRR*, abs/1804.02767, 2018. 5

[33] Vito Renò, Nicola Mosca, Massimiliano Nitti, Tiziana Dorazio, Donato Campagnoli, Andrea Prati, and Ettore Stella. Tennis player segmentation for semantic behavior analysis. In *IEEE International Conference on Computer Vision (ICCV) Workshop*, pages 718–725, Dec. 2015. 2

[34] Melike Şah and Cem Direkoğlu. Evaluation of image representations for player detection in field sports using convolutional neural networks. In *International Conference on Theory and Application of Fuzzy Systems and Soft Computing (ICAFS)*, pages 107–115, Cham, 2019. Springer International Publishing. 2

[35] Mamoru Saito, Katsuhisa Kitaguchi, Gun Kimura, and Masafumi Hashimoto. People detection and tracking from fish-eye image based on probabilistic appearance model. In *SICE Annual Conference 2011*, pages 435–440, Sep. 2011. 2

[36] Graham Thomas, Rikke Gade, Thomas B. Moeslund, Peter Carr, and Adrian Hilton. Computer vision for sports: Current applications and research topics. *Computer Vision and Image Understanding*, 159:3–18, 2017. Computer Vision in Sports. 2

[37] Paulius Tumas, Artūras Jonkus, and Artūras Serackis. Acceleration of HOG based pedestrian detection in FIR camera video stream. In *Open Conference of Electrical, Electronic and Information Sciences (eStream)*, pages 1–4, Apr. 2018. 2

[38] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *CoRR*, 2020. 3

[39] Tsaipei Wang, Chia-Wei Chang, and Yu-Shan Wu. Template-based people detection using a single downward-viewing fisheye camera. In *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pages 719–723, Nov. 2017. 2

[40] Tsaipei Wang and Chih-Hao Liao. People detection in downward-viewing fisheye camera networks using fuzzy integral. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–5, June 2019. 2

[41] Yukun Yang, Min Xu, Wanneng Wu, Ruiheng Zhang, and Yu Peng. 3D multiview basketball players detection and localization based on probabilistic occupancy. In *Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, Dec. 2018. 2

[42] Hui Zhang, Baojun Zhao, Linbo Tang, Jianke Li, and Jianke Li. Variational-based contour tracking in infrared imagery. In *International Congress on Image and Signal Processing*, pages 1–5, Oct 2009. 2

[43] Lijing Zhang, Yao Lu, Ge Song, and Hanfeng Zheng. RC-CNN: Reverse connected convolutional neural network for accurate player detection. In *Pacific Rim International Conference on Artificial Intelligence (PRICAI): Trends in Artificial Intelligence*, pages 438–446, Cham, 2018. Springer International Publishing. 2