

UNIVERSITY OF TECHNOLOGY SYDNEY
Faculty of Engineering and Information Technology

**MULTIPLE-CAMERA MULTIPLE-OBJECT 3D
LOCALIZATION IN SPORTS VIDEOS**

by

Yukun Yang

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

Sydney, Australia

2020

Certificate of Authorship/Originality

Yukun Yang declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Electrical and Data Engineering, Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature: *ProductionNote:
Signature removed prior to publication.*

Date: 15 / 03 / 2021

Acknowledgements

Throughout the writing of this thesis, I have received a great deal of support and assistance.

I would first like to thank my supervisor, Associate Professor Min Xu, whose expertise was invaluable in formulating the research questions and methodology. Your insightful feedback pushed me to sharpen my thinking and brought my work to a higher level.

I would also like to acknowledge my colleagues from my research projects at UTS for their excellent collaboration. I would particularly like to single out Dr. Yu Peng, Ruiheng Zhang, Wanneng Wu to thank you for your patient support and for all of the opportunities I was given to further my research. Especially for the object identification contributed by Ruiheng Zhang and object tracking by Wanneng Wu.

Finally, I would like to thank my parents for their support and encouragement throughout my study.

Yukun Yang
Sydney, Australia, 2020.

List of Publications

Journal Papers

- J-1. R. Zhang, L. Wu, **Y. Yang**, W. Wu, Y. Chen and M. Xu, "Multi-camera multi-player tracking with deep player identification in sports video," *Pattern Recognition*, vol. 102, p. 107260, 2020. (Published, see Chapter 5)
- J-2. **Y. Yang**, R. Zhang, W. Wu, M. Xu, "3D localization for multiple players in multiview sport videos with deep identification reasoning," *Pattern Recognition*. (Submitted for publication, see Chapter 6)

Conference Papers

- C-1. **Y. Yang**, M. Xu, W. Wu, R. Zhang, and Y. Peng, "3D multiview basketball players detection and localization based on probabilistic occupancy," in *2018 Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2018, pp. 1-8. (Published, see Chapter 4)
- C-2. **Y. Yang**, R. Zhang, W. Wu, Y. Peng, M. Xu, "Multi-camera sports players 3D localization with identification reasoning," in *Proceedings of the IEEE International Conference on Pattern Recognition*. (Published, see Chapter 6)

Contents

Certificate	ii
Acknowledgments	iii
List of Publications	iv
List of Figures	viii
Abbreviation	xiii
Notation	xiv
Abstract	xvi
1 Introduction	1
1.1 Background	1
1.2 Significance and Challenges	4
1.3 Research Objectives and Contributions	6
1.4 Thesis Organization	9
2 Literature Review	11
2.1 Multi-Camera Multi-Object Localization	11
2.1.1 Back-projection-based localization	12
2.1.2 Statistical modeling-based localization	14
2.1.3 Deep learning-based localization	16
2.1.4 RGB-D and point clouds-based localization	18
2.2 Person Re-Identification for Sports Players	22

3	Data Collection and Problem Formulation	25
3.1	Sports Video Recording and Data Preparation	25
3.1.1	Temporal synchronization for sports videos and image sequences	26
3.1.2	Multi-camera arrangement and calibration	27
3.1.3	Sports video datasets collection	29
3.2	Problem Formulation for 3D Localization	30
3.2.1	Modeling the sports space	32
3.2.2	Statistical modeling for locations of sports objects	35
4	POM+CNN+IniSet Localization Method	36
4.1	2D Monocular Segmentation for Multiple Sports Players	38
4.2	IniSet for The Bayesian Iteration	40
4.3	Experimental Evaluation	42
4.3.1	Datasets and metrics	43
4.3.2	Results	45
4.4	Conclusion	50
5	PomID Localization Method	52
5.1	DeepPlayer Model	53
5.1.1	Cascade Mask-RCNN	55
5.1.2	Pose-guided partial feature embedding	57
5.1.3	Obtain players' identification	58
5.2	PomID Localization Scheme	59
5.3	Experimental Evaluation	61
5.3.1	Datasets and experimental configuration	62

5.3.2	Results	62
5.4	Conclusion	65
6	PIOM Localization Method	67
6.1	Multi-Dimensional Bayesian Model	70
6.2	Image&ID Model	72
6.3	The Posterior Probabilities	75
6.4	An Efficient Iterative Process	78
6.5	Experimental Evaluation	79
6.5.1	Datasets	79
6.5.2	Metrics and baselines	80
6.5.3	Algorithm implementation and configuration	82
6.5.4	Results	83
6.6	Conclusion	92
7	Conclusion and Discussion	94
7.1	Conclusion	94
7.2	Discussion	96

List of Figures

3.1	An example of the sport court and its multi-camera arrangement method, based on a basketball court.	27
3.2	An example of the marked distinguished points on the sport court, based on a football dataset. We take the points of the black and white grids as the distinguished points, and mark their 3D world coordinates and 2D image pixel coordinates for each camera.	29
3.3	An example of the original image frames of our collected STU basketball dataset. Here shows four out of eight camera views. The camera arrangement method is illustrated in Figure 3.1.	31
3.4	An example of the original image frames of our collected LH0716v2 youth football dataset. Here shows eight camera views. The camera arrangement method is illustrated in Figure 3.1.	31
3.5	Top view of the ground plane and the discretized grid cells, every square grid has the same width \mathcal{W} , the number of these grids is G . The number of cameras varies from different datasets.	33
3.6	The 3D world coordinate system with a cube. We use $(\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{Z})$ to denote the 3D coordinate of the cube.	33
3.7	Back-projection of the cube on location k in camera view c . The ground plane is shown inside the blue lines. The occupancy of a target is described as a rectangle at the corresponding location. Note that some cubes' back-projected rectangles may not be visible if the camera views don't include the field where the location belongs.	34

4.1	This figure is an illustration of the implementation of 2D monocular segmentation for multiple sports players. We use CNN-based object segmentation to generate clear and correct foreground masks for the sports players that are captured by multi-camera. The outcomes of this method contain sports players' segmentation masks and bounding boxes. Those masks are removed when the players are standing outside of the sports courts.	39
4.2	Illustration of the IniSet model with indicative values, localization results are improved by eliminating large-scale miss-detection.	42
4.3	An example of the original image frames of the APIDIS basketball dataset. Here shows seven camera views.	43
4.4	Examples of experimental results of our proposed method. The green masks show where the players are presented. The black rectangles represent localization with qualified probabilities, while the blue areas denote likely occupied locations.	47
4.5	Examples of experimental results of our proposed method, the left column is the implementation of the POM method, while the right side is ours. Different rows present for different frames selected from various periods. The green lines denote qualified localization with probabilities higher than 0.8, while the red lines represent locations that are less likely to be occupied, with probabilities ranging from 0.2 to 0.8.	48
4.6	Precision and Recall curves in terms of the different thresholds. The first column is Precision versus different thresholds of three methods. The second column demonstrates Recall various from different thresholds. Additionally the third column is the Precision-Recall curves of these three methods.	51

5.1	The architecture of the DeepPlayer model. This model consists of two parts: (1) the Cascade Mask-RCNN for coarse-grained player detection(Cascade Mask-RCNN-P) and fine-grained jersey number recognition(Cascade Mask-RCNN-J); (2) the player mask embedding into the deep representation using PoseID. Finally, the player identity is decided by the jersey number class, the team class, and the deep representation.	54
5.2	PoseBox construction. Given a mask, the player pose is estimated by OpenPose. PoseBox1 = torso + arms + legs; PoseBox2 = head + torso + arms + legs; PoseBox3 = head + arms + legs.	59
5.3	Overview of the PomID model. The input of the PomID model includes objects' segmentation masks and ID labels. The 3D localization algorithm processes the players with identified ID and the players with ambiguous ID, respectively, followed by a post-process procedure with a set of experimentally defined thresholds. The output is the final 3D locations with distinguished identities.	60
5.4	Illustration of the PomID 3D localization results. The results are back-projected onto the original image sequences. Different colors indicate different team classes, while the labels of numbers indicate the different identities of multiple players across frames.	63
5.5	Illustration of the APIDIS dataset in Camera 3 and 6 indicates that the proposed method can avoid identity switches among Player 11, Player 14, and Player 15 in the dashed box.	64
5.6	Illustration of our localization results on the STU dataset in Camera 1, 4, and 6. The blue dotted boxes show that our method avoids identity switch between Player 6 of the white team and Player 11 of the black team.	65

- 6.1 An overview of the PIOM 3D localization framework. Firstly, we use the DeepPlayer model (see Section 5.1) that consists of a Cascade Mask-RCNN model and a PoseID model to extract the sports players’ segmentation masks and identification labels at pixel-level. At the same time, with the 3D world coordinate settings for the sports space mentioned in Section 3.2, we introduce an Image&ID model and an image distance norm to fuse the multiview pixel-wise segmentation and ID labels together with their 3D spatial relations. The synthetic images link the occupancy probabilities with the visible and computable image pixels, while the synthetic ID modules associate the identification inputs from all camera views with accurate spatial coordinates. With our proposed PIOM 3D localization algorithms, we then obtain sports players’ 3D locations and their unique ID labels. The localization results are finally given as the probabilities of locations that are occupied by the specifically labeled players. As shown above, different colors refer to different ID labels. 69
- 6.2 An example of the synthetic unit image, synthetic image and synthetic average images. (a) A synthetic unit image at location k from camera view c , the black area represents the back-projected ground plane grid on that location. (b) indicates the synthetic image where $X_{1,3} = 1$ and $X_2 = 0$. (c) and (d) are 2 examples of the synthetic average images $\bar{A}_{k,\zeta}^c$, when q_k has multiple values: $q_2 = 0.6, q_3 = 0.8, q_4 = 0.2$. But in (c) $q_1 = 1$, while in (d) $q_1 = 0$. . . 73
- 6.3 An example of the synthetic ID module. Three different colors refer to 3 different ID attributes. The first column shows the three synthetic unit ID \mathcal{R}_k s with different ID attributes from 3 different views at location k . The second column is the identification inputs from these views that including ID attributes and pixel information. The third column represents the calculated synthetic ID module R_k . . 74

- 6.4 Illustration of our localization results implemented on LH0716v2 dataset. The red cube refers to the generated location, while the yellow label indicates the detected identification outcome. In row No. 5 and 6, different colors refer to unique ID labels. Row No. 5 illustrates the localization results in 3D space, while No. 6 shows the results on the ground plane of bird-eye view. 84
- 6.5 Illustration of our localization results implemented on LH0928 dataset. Three rows indicate back-projection of the localization results from frame 3361, 3401, and 3441 on camera 2, 3, 6, and 7. The red cube refers to the generated location, while the yellow label indicates the detected identification outcome. 86
- 6.6 Illustration of our localization results implemented on LH0716v2 dataset that can overcome some extreme negative conditions, such as extremely crowded scenes, full/partial body occlusion, and inaccurate 2D detection from tiny objects. The first and third columns show the original back-projection of our localization results, while the second and fourth columns are obtained by zooming the crowded scenes inside the green borders. 87
- 6.7 Precision/Recall curves under multiple threshold settings. Results are obtained by implementing both PomID and our PIOM method on the LH0716v2 dataset. (a) and (d), P/R curves under BV distances range from $200 \sim 2000mm$; (b) and (e), P/R curves under IoU_{2d} ratios range from $0.05 \sim 0.95$; (c) and (f), P/R curves under IoU_{3d} ratios range from $0.05 \sim 0.95$. It illustrates that our proposed method PIOM outperforms the previous method PomID generally. . . 89

Abbreviation

2D: Two-dimensional

3D: Three-dimensional

AP: Average Precision

BV: Bird-eye View

CNN: Convolutional Neural Network

FPN: Feature Pyramid Network

IniSet: Initialization Settings

IoU: Intersection over Union

K-NN: K Nearest Neighbour

MCMOL: Multiple Camera Multiple Object 3D Localization

MODA: Multiple Object Detection Accuracy

MODP: Multiple Object Detection Precision

PIOM: Probabilistic and Identified Occupancy Map

POM: Probabilistic Occupancy Map

RCNN: Region Based Convolutional Neural Network

RPN: Region Proposal Network

ToF: Time-of-flight

Nomenclature and Notation

$c/C/\mathbf{C}$: the index/amount/set of cameras.

G/\mathbf{G} : the number/set of discretized grids.

k the index of grids.

$(\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{Z})$: the center point coordinate, the width, and the average height of a 3D cube, respectively.

$(X_{min}, Y_{min}, X_{max}, Y_{max})$: 4-element 2D coordinate of the rectangle.

t : timestamp.

$\mathbf{I}_t = \{I_t^1, I_t^2, \dots, I_t^C\}$: image sequence from camera \mathbf{C} with timestamp t .

$\mathbf{B} = \{B^1, B^2, \dots, B^C\}$: the information that is processed from the synchronized image frames from cameras \mathbf{C} .

$\mathbf{X} = \{X_k | k \in \mathbf{G}\}$: the set of Boolean random variables where X_k represents the presence and absence of a location k .

$\mathbf{Y} = \{Y_k | k \in \mathbf{G}\}$: the set of discrete random variable where Y_k represents the index of the identity of a location k .

\mathcal{A}_k^c : the synthetic unit image at location k in camera c .

A^c : the synthetic image in camera c .

$\overline{A}_{k,\zeta}$: the synthetic average image.

\mathcal{R}_k : the synthetic unit ID at location k .

R_k : the synthetic ID module.

q_k : the marginal probability at location k , also know as the posterior probability.

ε_k : the prior probability at location k .

i : index of the proposals.

g_i^c : groundtruth of the proposal region.

p_i^c : predicted classification of the proposal region.

p_i^l : predicted vector representing the offset between the i th proposal and its corresponding groundtruth bounding box.

g_i^l : the true offset value.

g_i^m : the groundtruth mask of the proposal region.

p_i^m : the predicted mask of the proposal region.

\mathcal{L}_{cls} : the loss of team classification.

\mathcal{L}_{loc} : the loss of player bounding box regression.

\mathcal{L}_{mask} : the loss of player mask.

ABSTRACT

MULTIPLE-CAMERA MULTIPLE-OBJECT 3D LOCALIZATION IN SPORTS VIDEOS

by

Yukun Yang

Sports video analysis and object 3D detection are extensively studied problems in computer vision. As one of the most important scenarios of object detection in 3D, multiple-camera multiple-object 3D localization (MCMOL) in sports videos has recently drawn much attention in the research community due to the growing trend of object detection from monocular to multiview, i.e., from 2D to 3D.

Due to heavy occlusion in crowded sports scenes and high-speed moving targets in sports games, MCMOL for sports objects tends to be extremely challenging. Existing solutions generally apply foreground extraction as input, design statistical or Convolutional Neural Network (CNN) models commonly to all visible targets to obtain objects' coordinates and/or location encoding. However, ambiguous foreground masks and heavy occlusion limit their performance by a large margin. Moreover, the obtained coordinates cannot be associated or retrieved back to the particular objects. There is no one-to-one relationship between the outcomes and the objects to be detected. Thus, the false-positive and false-negative rates increase.

To deal with the above-mentioned issues, in this thesis, we conduct comprehensive studies about the MCMOL problems in sports videos. Due to the challenges mentioned above, we develop three multi-camera multi-object 3D localization approaches that provide accurate, reliable, and distinguishable results. Firstly, we apply Convolutional Neural Network with Initialization Settings over the Probabilistic Occupancy Map (i.e., POM+CNN+IniSet). This approach applies CNN-based monocular segmentation jointly on multiple cameras and develops an indicative

parameter initialization scheme for the Bayesian iteration model. Afterward, we propose the POM with Identification (PomID) method and introduce the Deep-Player model including a Cascade Mask-RCNN model and a pose-guided partial feature embedding to conduct segmentation and identification simultaneously for multiple players. This method separately estimates locations for individuals with identified labels and the rest of the objects without specific identities. Finally, we propose the Probabilistic and Identified Occupancy Map (PIOM) method and develop an Image&ID model to mathematically describe the segmentation pixels and identification estimation as the likelihood probabilities. This method then creates a multi-dimensional Bayesian model to estimate the localization results as posterior occupancy probabilities with unique ID labels. Given the pre-defined prior probabilities, the Bayesian model is optimized by an efficient iterative convergence. Our work is the first attempt to take advantage of CNN-based object identification for object 3D localization applications.

Experimental results demonstrate that our proposed framework improves the localization performance by a large margin and outperforms the state-of-the-art in MCMOL sports video scenarios.

Chapter 1

Introduction

1.1 Background

Sports video analysis is a technique used to get information about moving objects from sports videos. Examples of this include gait analysis, sports replays, speed and acceleration calculations, and in the case of team or individual sports, task performance analysis [1, 2, 3]. The technique of sports video analysis usually involves a high-speed camera and a computer that has software allowing frame-by-frame playback of the video. Multi-camera sports video analysis is one of the most important applications in the field of video analysis and has received increasing interest in recent years. Various studies have been conducted in this area, including enhancing sports video broadcast [4, 5, 6], reconstructing 3D matches [7, 8, 9], and providing interactive content for audiences [10, 11, 12].

Object detection is an extensively studied computer vision problem, but most of the research has focused on object 2D prediction [13, 14, 15, 16, 17, 18, 19, 20, 21, 22]. Object 3D detection can capture an object's size, position, and orientation, leading to various applications in robotics, self-driving vehicles, image retrieval, and augmented reality. Object 3D detection has been applied in both monocular and multiview scenarios in recent years [23, 24, 25, 26, 27]. Approaches for object 3D detection process monocular or multiview image sequences as the input. Intermediate steps such as 3D world definition, camera calibration, monocular or multiview 2D detection, and localization algorithms designing are necessary and essential [28, 29]. These methods finally output the object's 3D world coordinates or discrete location

encoding as outcomes.

Being considered one of the most important scenarios of sports video analysis and object 3D detection, multiple-camera multiple-object 3D localization (MCMOL) [30] in sports videos has recently drawn much attention in the research community due to the growing trend of object detection from monocular to multiview, from 2D to 3D [31, 32, 33].

MCMOL for sports players aims to estimate the locations for all the players standing on the sports ground. The input of MCMOL methods usually contains image sequences extracted from multi-camera sports videos. Some other methods include more sensors such as stereo cameras and LiDAR scanners to solve this task. The outcomes of the MCMOL approaches typically consist of sports players' 3D bounding boxes, location coordinates, or location encoding.

3D localization of multiple sports players in multiview sports videos is quite different from the traditional scenarios because of the following characteristics: [34, 35]:

1. Multiple cameras applied in sports scenarios
2. The huge data volume of sports videos and images
3. High-speed moving objects
4. Complicated and changing background contents
5. Heavy occlusion in crowded sports scenes

Thus, 3D localization is more challenging because it applies object detection from 2D to 3D, from monocular to multiview. The 3D localization results of multiple sports players are often inputted into multi-object tracking algorithms that rely on

tracking-by-detecting strategies to optimize continuous trajectories for each sport player, respectively [36].

Due to heavy occlusion and rapid-moving players in crowded sports scenes, MCMOL for sports players tends to be extremely challenging. In order to tackle the task of MCMOL for sports players, most existing methods utilize the information across multiple views, which generally have overlapping fields of view captured from different orientations [37, 38]. Precise camera calibration is required to calculate and project between the image pixels and the corresponding 3D world points in the sports space. The calibration outcomes associate visual information from multiple views and link the image input with the 3D location coordinates that are generally considered the algorithms' output. [39, 40]. In order to keep the consistency of the 2D input evidence from different views, image sequences are extracted from the original sports videos according to a timestamp calculator, which ensures that images from different views with the same timestamp indicate the consistent sports game status. Some of these methods pre-process RGB images from different views and extract the moving players as foreground masks [41, 42, 43], which are lately inputted into the statistical models [44, 45] or CNN-based detection networks [46, 47, 48]. In comparison, some state-of-the-art methods apply different types of sensors such as ToF cameras and LiDAR to obtain RGB-D images [49, 32, 50, 51, 52] or point clouds [53, 54, 55, 56, 57], which can deliver accurate distance information from targets to the sensors.

Those methods mentioned above can generally achieve satisfactory precision and recall. However, traditional methods that use foreground extraction usually generate ambiguous segmentation masks, which increases the false-positive results. Besides, RGB-D or point-cloud-based methods are often limited by high costs of equipment [58, 59]. Moreover, all these approaches cannot provide distinguishable and unique outcomes among multiple objects. They cannot significantly increase

the true-positive rate and avoid false-positive detection.

1.2 Significance and Challenges

Multi-camera multi-object 3D localization is significant. MCMOL helps develop public transportation systems, industrial office security, and pedestrian protection applications for video surveillance systems. For object tracking tasks, it provides the most important input and significantly impacts the tracking performance. More importantly, MCMOL for sports players plays a crucial role in the research of multiple player tracking, sports event analysis, and sports event prediction.

Existing approaches for MCMOL in sports videos mentioned above have been widely studied and applied in many applications. However, ambiguous 2D extraction, tiny pixel blobs in remote views, high speed moving targets always limit their localization performance by large margins. In addition, depth images and point clouds require extremely high-cost equipment. More importantly, a critical influencing factor on MCMOL performance is the heavy occlusion in extremely crowded sports scenes such as basketball and football.

Meanwhile, current methods can only provide a set of estimated locations for all the visible players. This kind of outcome can not be associated or retrieved back to the particular player who is proved to be detected at the given location. Therefore, we explore the player identification in order to obtain distinguishable results that every estimated location is unique among each other. Given the localization results with unique player identities, we can significantly improve the localization performance by eliminating false-positive results, hence overcome the limitation caused by heavy occlusion. Besides, it has great potential to eliminate the negative impact caused by identity switches in some multi-target trackers using tracking-by-detecting schemes.

What is more, player identification is even more challenging in real-world sports videos. Unlike pedestrians [60] and vehicles [61, 62] that have relatively predictable motion patterns, sports players tend to confuse their opponents with abrupt moves in directions and unexpected changes in velocity. Meanwhile, compared with person re-identification, commonly used features for re-id, e.g., color and gait, become invalid in the scenario of player identification.

In order to analyze the current methods and address the existing problems aforementioned, we summarize the following challenges with respect to the MCMOL tasks for sports players:

1. Huge data volume of sports videos and image sequences, temporal synchronization of multi-camera sourced image frames
2. Inaccurate sports space calibration, installation, and arrangement of multiple cameras.
3. Ambiguous sports players segmentation inputs, changing background contents of sports courts.
4. Player's jersey number encounters severe deformation due to the movement. Low resolution and variant image size also make the jersey number difficult to recognize.
5. Player's similar appearance due to the similar uniform, body shape variation, erratic motion, spectator interference, and the illumination variation makes it difficult to track and identify players reliably.
6. Player's heavy occlusion caused by extremely crowded sports scenes widely exists and brings high-level false-positive and false-negative rates. Ambiguous extraction input, tiny pixel blobs in remote views, rapid-moving targets also limit the localization performance.

7. Player's locations estimated by existing methods cannot be associated and retrieved back to the particular player. These types of localization outcomes are not distinguishable and unique, therefore generate more false-positive results and more identity switches for the further trackers who intend to take benefits from 3D localization as input.

We have conducted comprehensive research about the MCMOL problems for sports players and illustrated the proposed methods in Chapter 4, Chapter 5, and Chapter 6 to solve these challenges. Challenge No. 1 and 2 are discussed in Chapter 4, challenge No. 3 is solved in Chapter 5, while challenge No. 4 to 7 are studied in Chapter 6.

1.3 Research Objectives and Contributions

As we have concluded the challenges for the problem of MCMOL in sports videos, therefore, we illustrate the research objectives for this thesis as follows:

1. To eliminate the unclear input and ambiguous foreground masks for the localization algorithms and enrich the prior evidence for the localization algorithms from the fisheye camera detection, we implement CNN-based segmentation and apply prior parameter initialization from fisheye cameras.
2. To overcome heavy occlusion in crowded scenes, reduce false-positive and false-negative rates, and to obtain distinguishable and unique location results, we introduce object segmentation and identification into the localization algorithms and conduct computation for individual objects and uncertain objects separately.
3. To overcome heavy occluded sports scenes, overcome special sports conditions, obtain distinguishable and unique localization results that can be retrieved

back to the particular object, and establish an integral and robust localization framework, we introduce object segmentation and identification jointly into a multi-dimensional Bayesian localization model.

Existing approaches cannot comprehensively solve those challenges mentioned in Section 1.2. There is a lack of an integral and reliable solution for sports players' MCMOL tasks in both academia and industry. Therefore it is desirable to propose a framework that studies multi-camera multi-object 3D localization for sports players and can provide accurate, reliable, and distinguishable 3D localization results.

Considering the problems mentioned above and the limitation caused by the previous solutions, in this thesis, we conduct a comprehensive research about the MCMOL problems for sports players and proposed three stages of frameworks corresponding to the research objectives concluded above in order to tackle the tasks and challenges mentioned in Section 1.2. Our main contributions are concluded as follows.

1. **POM+CNN+IniSet** 3D localization method (see Figure 4.2). This method applies the CNN-based monocular segmentation jointly on multiple cameras in order to remove the ambiguous results generated by the ordinary background extraction and eliminate missed foreground masks. Additionally, we develop a generic Bayesian model with an indicative parameters initialization scheme for the localization iteration from the fisheye detection input. This approach enriches the localization system's input information, removes undesired segmentation masks, improves the precision and recall of the localization outcomes, and boosts the 3D localization performance.
2. **PomID** 3D localization method (see Figure 5.3). This method proposes a DeepPlayer model including a Cascade Mask-RCNN model and a pose-guided partial feature embedding to conduct object segmentation and identification

for multiple sports objects. The DeepPlayer model produces both the individuals' foreground masks and their identities, which are treated as the given evidence for the 3D localization algorithms. This method then separately estimates the likely location for each player who has a certain and correct identity and jointly calculates the results for the rest targets without ID labels. Final outcomes are then refined by a set of reasonable constraints. This approach includes multiple objects' identities as evidence to estimate the likely occupied locations, making the localization results distinguishable and unique to be associated with the particular objects. This method can accurately locate multiple objects and effectively avoid identity switches for multiple-object detection and tracking tasks. To our best knowledge, this is the first attempt to introduce object identification into MCMOL approaches.

3. **PIOM** 3D localization method (see Figure 6.1). This method firstly takes 2D segmentation and identification from all camera views as input. It then develops an Image&ID model to visually describe the status of an object's presence and identity in a specific location. It associates the binary pixel input with the mathematical format of occupancy and identification probabilities. Afterward, we develop a multi-dimensional Bayesian model and construct a loss function as the $K - L$ divergence between an estimated probability distribution and the true posterior probability. The prior probabilities are initialized at the beginning of the iteration, while the likelihood probabilities are approximated by the normalized image distance. Finally, an efficient iterative process is designed to minimize the loss function and obtain the optimal solutions. The PIOM method generates accurate locations with correct identities for every object that is visible in the detection space. As the localization outcomes are unique and distinguishable for each player, this method can effectively overcome the challenges mentioned above. Meanwhile, it still keeps excellent

performance in some extreme conditions such as super heavy occlusion, partial body occlusion, and high moving speed body gestures.

1.4 Thesis Organization

In this section, we introduce a brief overview of our thesis. This thesis is organized as follows:

- *Chapter 2:* In this chapter, we present a survey of the existing approaches for 3D multiview object localization and object identification tasks, especially for multiview sports video scenarios. We also investigate the state-of-the-art methods and their applications.
- *Chapter 3:* In this chapter, we introduce the sports video recording and data preparation, including temporal synchronization for sports videos and image sequences, multi-camera arrangement and calibration, and five sports video datasets we collected and processed. Then we present the problem formulation for 3D multiview multiple-object localization, which includes the sports space modeling and statistical modeling for multi-object localization.
- *Chapter 4:* In this chapter, we illustrate our proposed POM+CNN+IniSet 3D localization method, including multi-camera 2D segmentation and initial setting for Bayesian iteration. Then we conduct the experiments based on the APIDIS dataset and compare the results with the baseline POM.
- *Chapter 5:* In this chapter, we present our proposed PomID 3D localization method, which includes the DeepPlayer model and PomID localization scheme. Afterward, we conduct the experiments based on the public dataset APIDIS and our collected dataset STU and evaluate the performance of our proposed method.

- *Chapter 6:* In this chapter, we introduce our proposed PIOM 3D localization method, including the multi-dimensional Bayesian model, the Image&ID model, the calculation of the posterior probabilities, and an efficient iterative process. Then we conduct the experiments based on two of our collected football datasets LH0716v2 and LH0928, and compare the outcomes with the baseline POM and two of our proposed methods mentioned above.
- *Chapter 7:* In this chapter, we present a brief summary of the thesis contents and our contributions to the MCMOL tasks in sports videos. Discussion for future work is presented as well.

Chapter 2

Literature Review

In this chapter, we provide an introduction to the related work for the MCMOL tasks. Firstly we present a comprehensive investigation about the existing MCMOL methods in Section 2.1, which can be roughly divided into four categories: back-projection-based localization, statistical modeling-based localization, deep learning-based localization, and depth image and point clouds-based localization. The back-projection-based localization and statistical modeling-based localization are the most traditional approaches for the MCMOL tasks. Deep learning-based localization and depth image and point clouds-based localization are two types of the most recent studies.

Additionally, as we develop person re-identification models for sports players in Chapter 5 and 6, we review some person re-identification methods applied in sports video scenarios in Section 2.2.

2.1 Multi-Camera Multi-Object Localization

Traditional methods that address multi-camera multi-object 3D localization are RGB image-based back-projection and statistical modeling-based algorithms.

These methods generally utilize the information across multiple views, which have overlapping fields of view captured from different orientations. Precise camera calibration is required to calculate and project between the image pixels and the corresponding 3D world points in the detection space. The calibration outcomes associate visual information from multiple views and link the image input with the

3D location coordinates that are generally considered the algorithms’ output. In order to keep the consistency of the 2D input evidence from different views, image sequences are extracted from the original videos according to a timestamp generator, which ensures that images from different views with the same timestamp indicate consistent content.

These methods generally pre-process the RGB images from different views and extract the moving targets such as pedestrians or sports players to generate foreground masks. The foreground masks are typically considered the most critical input of the 3D localization algorithms. These approaches can be roughly divided into two groups.

2.1.1 Back-projection-based localization

The first group of those traditional methods is called back-projection-based localization. Those solutions use foreground subtraction and accurate camera calibration data as input. Localization algorithms are developed to back-project the foreground masks into one or several reference planes [41, 43, 63, 64, 65]. The reference planes are usually selected to be parallel with the ground plane of the detection space. With back-projected blobs including intersecting parts on these planes, models are constructed to analyze interconnected masks and occlusions. Localization results are then obtained based on geometric computations of the masks on the reference planes.

Some methods such as [41] back-projected the foreground masks from multi-view onto a reference plane with head level, using the head segmentation to locate pedestrians’ coordinates. Khan and Shah [43] selected several reference planes with multiple heights, developed a planar homographic occupancy constraint that fuses foreground masks from multiview to resolve occlusion.

Ge and Collins [63] used a Gibbs point process stochastic process to model the

generation of multiview images of random crowd configurations. The optimal crowd configuration was estimated by sampling a posterior distribution to find the MAP estimate for which the model best fits the image observations.

Utasi and Benedek [64] assumed that the scene is monitored by multiple calibrated cameras and the extracted foreground masks are available. The foreground pixels were projected on the ground and multiple parallel planes. This method extracted two similar pixel-level features in each 2D position: one on the ground plane and one on each head plane. The extracted features were used in a stochastic optimization process with geometric constraints to find the optimal configuration of multiple people.

Lo et al. [65] developed a vanishing point-based line sampling technique for dense people localization in real-time and multiple camera tasks. For each camera view, they projected sample lines originated from a vanishing point of the foreground objects on the ground plane. Ground regions containing a high density of projected lines were then used to find people locations.

Hsu et al. [66] introduced a torso-high reference plane because in general the torso part is more intact and stable than the other parts of a human body and thus can predict potential people locations more reliably. They then proposed a bit-wise-operation scheme to predict people locations at the intersection regions of foreground line samples from multiview. Rule-based validation was then used to obtain and visualize people’s locations on a real-world plane.

Those methods typically achieve acceptable performance with high efficiency when the scene is not crowded. However, their performance is limited for crowded and complex scenarios such as pedestrians and sports players.

2.1.2 Statistical modeling-based localization

The second group of those approaches mentioned above is statistical modeling-based localization. Those solutions consider foreground masks and camera calibration data as input as well. In contrast, those approaches directly model the objects' locations and occlusion using statistical algorithms such as Bayesian models [34, 42, 67, 44, 45, 68]. The iterative process is commonly used to intend to achieve optimal occupancy probabilities.

For example, Fleuret et al. [42] (POM) extracted multiview foreground masks as input, constructed a Bayesian model to estimate the occupancy probabilities as the posterior probabilities. The prior probabilities of the Bayesian model are configured at the beginning of the iterative process. Because the generative model explicitly calculates the occlusion, POM is robust and performs typically well. However, it relies on foreground masks as the only input, which is not discriminative enough when the objects' density increases. Thus it often produces false-positive outcomes. Additionally, its localization results cannot be associated and retrieved back to the particular players. Therefore it would generate more false-positive locations.

Peng et al. [67] proposed a multiview Bayesian network model to detect pedestrians from multi-camera surveillance videos. They discretized the ground plane in a predefined set of locations and modeled the potential occlusion relationship of all locations in all views. A set of Boolean parameters were then estimated to denote whether a pedestrian occurs at the corresponding location.

Based on their previous research, Peng et al. [44] then developed a multi-camera pedestrian detection approach with a multiview Bayesian network model. They used the model to describe both the occlusion relationship and the homography correspondence. This approach is robust because it can effectively remove phantoms from pedestrian candidates. However, this method constructs the Bayesian model

for all pedestrian candidates and their occlusion relationship in all views, making it considerably tricky and time-consuming to calculate the highest probabilities.

Yan et al. [68] warped each foreground intersection region back to the original camera view and associated the region with a candidate box of the average size of pedestrians at the location. They then calculated a joint occupancy likelihood for each intersection region. Afterward, essential candidate boxes were identified first, each of which covered at least a part of the foreground that is not covered by another candidate box. The non-essential candidate boxes were selected to cover the remaining foregrounds in the order of their joint occupancy likelihoods.

Klinger et al. [69] developed a joint probabilistic data association framework for the assessment of similarities between detection and tracked targets. They proposed a dynamic model which is based on Gaussian Process Regression. They formulated a new co-variance function taking the spatial distance and the angular displacement of two trajectories into account. The output of the co-variance function was used as a measure for the interaction between pedestrians. The 2D image coordinates were related to the world coordinates by the col-linearity equations.

Rubino et al. [45] proposed a method that uses 2D detection to recover objects' 3D position and occupancy. They formulated the problem as estimating a quadric in 3D given a set of 2D ellipses fitted to the object detection bounding boxes in multiview. A non-linear optimization scheme was devised to cope with the possible ill-conditioning of the problem. However, this approach cannot deal with heavy occlusion such as crowded pedestrians and sports games. Especially when the objects are tiny on the camera views, the performance is severely limited.

The statistical modeling-based methods typically model the objects' locations and their occlusion relationship based on statistical models such as the Bayesian model. They usually achieve good performance for multiview tasks without high-

level density objects. However, the processing time for their localization algorithms is considerably extended, and the calculation is very tricky.

2.1.3 Deep learning-based localization

As CNN-based methods have achieved significant progress in the area of Computer Vision, especially in the field of Object Detection and Segmentation, a significant number of researchers have paid attention to the Deep Learning methods to address the multi-camera multi-object localization and tracking issues. Since the year 2016, many deep learning-based approaches have been proposed, including network construction and data training [30, 32, 46, 48, 70, 71]. Most of these methods construct two networks that include a 2D monocular detection network and a 3D multiview localization network. The 2D monocular detection network is implemented to detect 2D information of objects, which usually include segmentation masks, bounding boxes, and image view coordinates. Afterward, the 3D multiview localization network is developed to fuse the results of the 2D detection network from multiview and estimate the 3D coordinates of multiple objects. The localization results are typically produced as the exact 3D coordinates or the location encoding.

For example, Zhang et al. [72] investigated the perfect single frame detector for pedestrian 3D localization. They studied the impact of training annotation noise on the detector performance, analyzed failure cases of top performance pedestrian detectors, and diagnosed what should be changed to further push performance. They addressed the high false-positive rate by improving the training set alignment quality by manually sanitizing the Caltech training annotations and using algorithmic means for the remaining training samples.

Chen et al. [70] proposed a 3D object detection approach for autonomous driving. This method generated a set of candidate class-specific object proposals that

are then run through a standard CNN pipeline. They proposed an energy minimization approach that scores each object candidate box via several intuitive potentials encoding semantic segmentation, contextual information, size and location priors, and typical object shape.

Mousavian et al. [46] proposed a method that estimates the pose and the dimensions of an object's 3D bounding box from a 2D bounding box using the constraints provided by projective geometry and estimates of the object's orientation and size regressed using a deep CNN. They regressed the orientation and object dimensions before combining these estimates with geometric constraints to produce a final 3D pose.

Bagautdinov et al. [71] proposed a CNN that simultaneously solves multi-person detection, individual action recognition, and collective activity recognition. This method relies on joint multi-scale features that are shared among all the tasks. It used a probabilistic inference scheme to refine the detection hypotheses.

Baqué et al. [30] developed an architecture that combines CNN and Conditional Random Fields (CRF) to model the ambiguities and the potential occlusion explicitly. It produced probabilities of presence on the ground plane, which can be linked into full trajectories.

Kim et al. [73] proposed a deep learning network composed of a detection network and a localization network. An attentional pass filter was introduced to pass a detection candidate that may be a pedestrian. The optimal results were achieved through the min-cost network flow approach.

Chavdarova and Fleuret [74] fine-tuned one of their previous object detection network on monocular pedestrian detection and combined several instances of the early layers of this network into a multiview deep network whose outer layers are trained for multiview appearance-based joint detection on a relatively smaller multi-

camera dataset.

Wang et al. [47] proposed a multi-level important salient feature detection approach based on data-adapting convolution filters and a data-driven algorithm. They aggregated the important saliency map with color features to formulate an appearance model. They then developed a support-vector-machine-based incremental learning method by using modified regularization terms to build and update the appearance model online and recognize the object based on a classification method. The proposed method can effectively discriminate new target objects that were never learned in the primary model and simultaneously improve the matching accuracy of old objects.

However, none of those methods have been implemented based on sports video datasets. Currently, the suitable datasets that are publicly available are not sufficient to support the implementation of data training and experiments for CNN-based methods. Moreover, the localization performance is still limited when the multiview 2D detection results are fused to fit the 3D localization network.

2.1.4 RGB-D and point clouds-based localization

Since the year 2017, researchers have begun to develop approaches using depth images and LiDAR point clouds to solve multi-camera multi-object 3D localization. The most state-of-the-art approaches that apply RGB-D images and point clouds are proposed to tackle the problems of autonomous driving and pedestrian detection [31, 53]. The depth image is an image or image channel that contains information relating to the distance of the surfaces of scene objects from a viewpoint. The point cloud is a set of data points in space. Point clouds are generally produced by 3D scanners or by photogrammetry software, which measures many points on the external surfaces of objects around them. These methods that use RGB-D or point clouds can acquire more accurate distance information of the objects from the

viewpoints. They generally combine the traditional RGB images with the depth dimension or with LiDAR point clouds to extract hand-crafted features, using R-CNN to propose 2D RoI and regress the objects' coordinates. The most recent studies were proposed to directly model the 3D locations using discretized 3D voxel grid representation other than 2D pixels with depth [54].

For example, Chen et al. [75] solved the problem of generating high-quality 3D object proposals in the context of autonomous driving as minimizing an energy function encoding object size priors, ground plane as well as several depth informed features that reason about free space, point cloud densities, and distance to the ground.

Based on their previous studies, Chen et al. [31] afterward proposed a sensory-fusion framework that takes both LiDAR point clouds and RGB images as input and predicts oriented 3D bounding boxes. They developed a network that generates 3D candidate boxes from the bird-eye view representation of 3D point clouds. A deep fusion scheme was introduced to combine region-wise features from multiview and enable interactions between intermediate layers of different paths.

Engelcke et al. [53] developed an approach to detect objects natively in 3D point clouds using CNN. They exploited feature-centric voting to build CNNs to detect objects in 3D point clouds without projecting the input into a lower-dimensional space first or constraining the search space of the detector. This enables the CNNs to learn high-capacity and non-linear models while providing constant-time evaluation at test-time.

Li [76] used a 3D fully convolutional network (FCN) to enhance the performance of object detection in the point cloud. The method they proposed detects objects and estimates oriented object bounding boxes in an end-to-end manner.

Qi et al. [50] extracted the 3D bounding frustum of an object by extruding 2D

bounding boxes from image detectors. Within the 3D space trimmed by each of the 3D frustums, they consecutively performed 3D object instance segmentation and amodal 3D bounding box regression using two variants of Point-Net. The 3D mask of the object of interest was predicted by the segmentation network, and the amodal 3D bounding box was estimated by the regression network.

Xu and Chen [77] proposed a framework for 3D object detection by estimating the object class, 2D location, orientation, dimension, and 3D location based on a single monocular image in an end-to-end fashion. A region proposal network was utilized to generate 2D proposals in the image. Two more branches were added for jointly learning of orientation and dimension. For 3D object dimension, typical sizes made up of length, width, and height were accessed by analyzing the training labels for each class.

Xu et al. [78] developed a deep network for 3D object box regression from images and sparse point clouds. The network consisted of an off-the-shelf CNN that extracts appearance and geometry features from input RGB image crops, a variant of PointNet that processes the raw 3D point cloud, and a fusion sub-network that combines the two outputs to predict 3D bounding boxes. The network then used a learned scoring function to select the best prediction.

Zhou and Tuzel [54] proposed a 3D detection network that unifies feature extraction and bounding box prediction into a single-stage, end-to-end trainable deep network. This network divides a point cloud into equally spaced 3D voxels and transforms a group of points within each voxel into a unified feature representation through the voxel feature encoding layer. The point cloud is encoded as a descriptive volumetric representation, which is then connected to an RPN to generate detection.

Ku et al. [79] proposed a neural network architecture using LiDAR point clouds and RGB images to generate features that are shared by two subnetworks: a re-

gion proposal network and a second stage detector network. The region proposal network used an architecture capable of performing multimodal feature fusion on high-resolution feature maps to generate reliable 3D object proposals for multiple object classes in road scenes. They used a 3D bounding box encoding that conforms to box geometric constraints, allowing for higher 3D localization accuracy. The proposed neural network architecture exploited 1×1 convolutions at the region proposal network stage, allowing high computational speed and a low memory footprint.

Wang and Jia [80] developed a method termed Frustum ConvNet (F-ConvNet) for amodal 3D object detection from point clouds. This framework first generates a sequence of frustums to group local points. F-ConvNet aggregates point-wise features as frustum-level feature vectors and arrays these feature vectors as a feature map for the use of its subsequent component of a fully convolutional network, which spatially fuses frustum-level features and supports an end-to-end and continuous estimation of oriented boxes in the 3D space.

Li et al. [32] developed a 3D object detection method for autonomous driving by fully exploiting the sparse and dense, semantic and geometry information in stereo imagery. This method extends Faster R-CNN [17] for stereo inputs to simultaneously detect and associate objects in the left and right images. Extra branches were added to predict sparse key points, viewpoints, and object dimensions, which are combined with 2D left-right boxes to calculate a coarse 3D object bounding box. The accurate 3D bounding box was recovered by a region-based photo-metric alignment using the left and right RoIs.

However, the RGB-D and point clouds-based approaches have not been developed to tackle the issues of sports player detection so far. Most scenarios that apply RGB-D and point clouds are autonomous driving and pedestrian detection, which typically use a monocular camera or a single LiDAR device. For sports video scenar-

ios, multiple cameras and devices are essential to the localization implementation. Moreover, the LiDAR devices are usually too expensive for the sports video datasets. The localization performance is still restricted because of the low accuracy of depth estimation and high error-sensitive point clouds.

Unlike other approaches mentioned above, in this thesis, we take advantage of the Bayesian model and CNN-based object detection to output discriminative segmentation masks and precise location encoding. Furthermore, we apply CNN-based object detection and person re-identification simultaneously to the sports players and fuse this information from multiview into a multi-dimensional Bayesian framework that explicitly models the occlusion, the players' coordinates, and their identities.

Different from other solutions that use CNN to extract feature representation and model the 3D bounding boxes, our proposed method is the first attempt to leverage sports players' identities to obtain distinguishable locations. Thus, the output of our developed method is also different from the others.

2.2 Person Re-Identification for Sports Players

Person re-identification algorithms for sports players are concentrated on a close-up single camera view and multiple camera views. Under the close-up camera views, players can be identified by jersey number recognition [81, 82, 83] and face recognition [84].

For jersey number recognition, these approaches try to directly recognize the jersey numbers followed by character recognition without detecting the number regions. Ye et al. [82] employed a K-NN (K Nearest Neighbour) classifier with the Zernike moment features to detect jersey numbers for sports players. Gerke et al. [81] first introduced CNNs into football jersey number recognition. Without any

character detectors, Li et al. [83] developed CNN models to classify jersey numbers on the images where the players are detected. The method in [85] combined the textual cues with the visual face information to try to identify players.

Somewhat against the trend, in terms of multiple cameras, facial information is often limited. Lu et al. [86] first attempted to track and identify basketball players by recognizing the entire body instead of the face or jersey numbers. They designed a sports player’s appearance representation by low-level hand-crafted features, including scale-invariant feature transform, maximally stable extremal regions, and color histograms.

Differently, we leverage deep convolutional features guided by pose estimation to model the players’ representation. Recently, Gerke et al. [81] treated jersey number recognition as an image classification task using deep convolutional neural networks. They directly cropped the top half of the sports images as the jersey number regions. In [87], players’ jersey numbers and group information were used to associate tracklets of the same player. In this thesis, we develop a coarse-to-fine-grained deep convolutional neural network to simultaneously detect and recognize the jersey numbers and team classes for the sports players. Senocak et al. [88] used convolutional neural network features to represent the player regions and formulate the identification task as a classification problem. Compared with this method, we pay more attention to how to find the difference among multiple players.

Some other studies perform sports player identification by using the position of players. In [89], the players’ location information is utilized as the spatial constellation apart from the jersey number recognition. As players’ trajectories are known, the problem is formulated as an assignment problem. Lu et al. [86] leveraged both detection and tracking to build a conditional random field model for all the players. In contrast, we do not perform the player identification using any temporal infor-

mation, which means that the identification process can be performed in any frame at any moment.

Chapter 3

Data Collection and Problem Formulation

3.1 Sports Video Recording and Data Preparation

Being considered as an important application of video processing, multiple-camera multiple-object 3D localization (MCMOL) has recently drawn much attention in the research community due to the high demand for multiview sports video analysis.

In order to solve the problems of MCMOL for sports objects, most existing methods utilize the information across multiple views, which generally have overlapping fields of view captured from different orientations. Precise camera calibration is required for each camera to calculate and project between the image pixels and the corresponding 3D world points in the sports space. The calibration outcomes can associate visual information from multiple views and link the captured images from different views with the particular 3D coordinates of objects. In order to keep the consistency of the video content from different views, image sequences are extracted from the original sports videos according to a timestamp calculator. The timestamp calculator can record one accurate timestamp for each frame, ensuring that frames from different views with the same timestamp indicate the consistent sports game status.

Thus, sports video acquisition, temporal synchronization for sports image sequences, multi-camera arrangement, and camera calibration are critical for solving the problems of MCMOL for sports objects.

In this section, we introduce a temporal synchronization scheme for sports videos

and image sequences, a multi-camera arrangement and calibration method with an example implementation for a football court, and an introduction to the datasets we have collected and processed for our sports video research.

3.1.1 Temporal synchronization for sports videos and image sequences

For the 3D localization of multiple objects on the sports court, we acquire the original sports videos from several wide-angle high-resolution video cameras. We use a timestamp calculator to unify the filming time and duration for all installed cameras. The timestamp calculator controls the starting time and ending time jointly for all cameras. Furthermore, it will record the current time of the original video streaming, which makes it easy to double-check if the image sequences from different cameras remain consistent.

From the original sports videos to the corresponding image sequences, we extract image frames from all camera videos with a unified setting, including fps, resolution, starting timestamp, and ending timestamp. However, errors are inevitable when we extract the image frames from sports videos. These errors would cause inconsistent content for different views. For rapid-moving sports scenarios, even extremely slight errors would cause severe negative results.

Thus, we only select the image sequences that indicate the sports game was ongoing and abandon those frames when the sports game was in a pause or was having half-time. Additionally, we collect the accepted image sequences into several groups, and we call them periods. Each period keeps the same amount of image sequences from each camera view. The starting frame and ending frame remain consistent among all camera views.

3.1.2 Multi-camera arrangement and calibration

In order to appropriately capture the overlapping fields of view of the sports court, multiple cameras need to be arranged around the court with appropriate settings of position and height. In our collected datasets, we installed eight cameras around the court with a typical player height ($1.8m$). Note that all the cameras should be arranged to cover all visible areas on the sports court. Otherwise, if a player is presented on any invisible location, the localization algorithm cannot detect his position.

An example of the multi-camera arrangement based on a basketball court is shown in Figure 3.1.

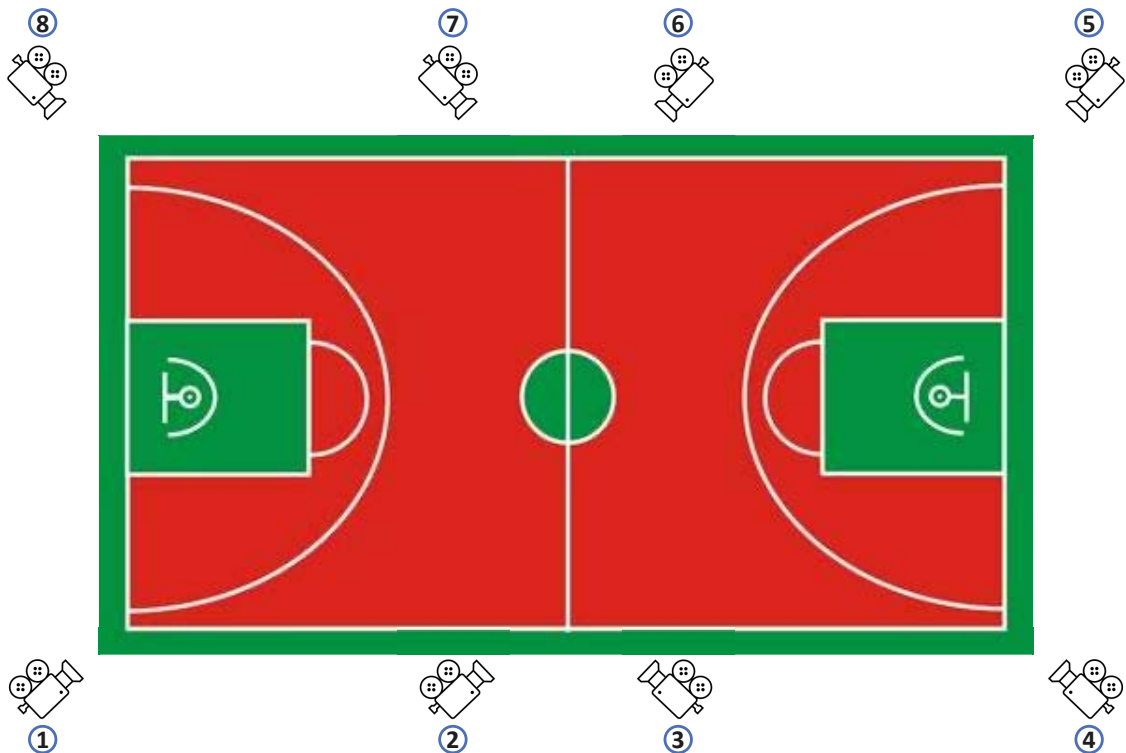


Figure 3.1 : An example of the sport court and its multi-camera arrangement method, based on a basketball court.

Precise camera calibration is required for each camera to calculate and project

between the image pixels and the corresponding 3D world points in the sports space. The calibration outcomes can associate visual information from multiple views and link the captured images from different views with the particular 3D coordinates of objects.

Generally speaking, camera calibration consists of intrinsic parameter calibration and extrinsic parameter calibration. Currently, high-quality cameras manufactured by the industry have fixed intrinsic parameters that vary from different brands and models. However, in order to accurately obtain the intrinsic parameters, we apply a camera calibration toolbox based on [39] to conduct the estimation of the intrinsic parameters.

Regarding the extrinsic parameters, they are associated with the spatial relationship between cameras and sports courts. If the position, orientation, or height of a camera is modified, the extrinsic parameters of that camera are then consequently changed. Thus, if the extrinsic parameters are calibrated, the position, orientation, and height of a camera's arrangement cannot be changed.

To estimate the extrinsic parameters, we mark the 3D coordinates of a number of distinguished points on the sports court, then capture one image for each camera. Afterward, we identify the 2D image pixel coordinates of these marked points for every camera and pair the 2D coordinates with their corresponding 3D world coordinates. Finally, we use these pairs of coordinates to calculate the extrinsic parameters for every camera.

An example of the marked distinguished points on the sports court is shown in Figure 3.2, which is based on a football dataset.

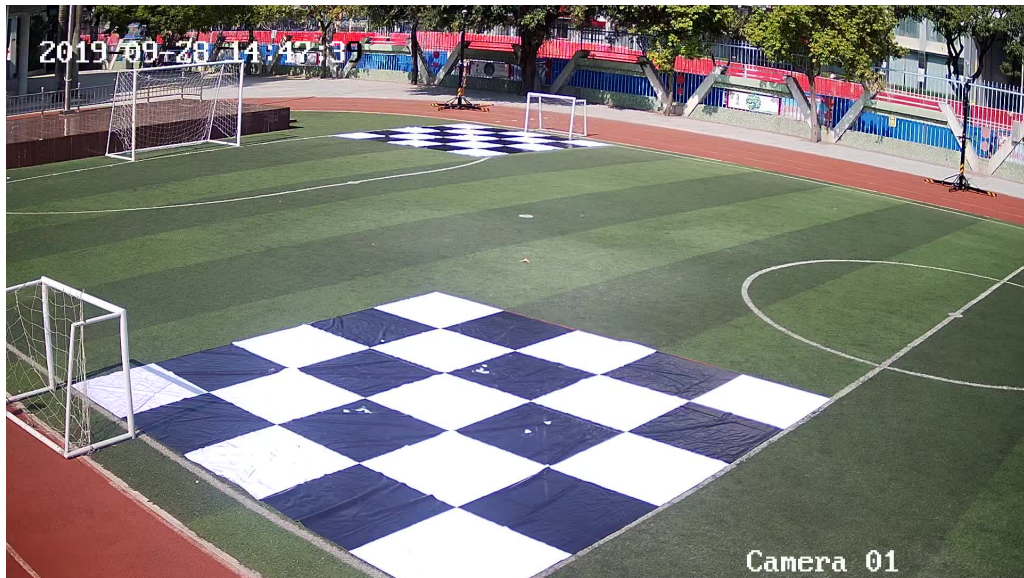


Figure 3.2 : An example of the marked distinguished points on the sport court, based on a football dataset. We take the points of the black and white grids as the distinguished points, and mark their 3D world coordinates and 2D image pixel coordinates for each camera.

3.1.3 Sports video datasets collection

In order to implement our proposed 3D localization method, we have collected and processed five sport video datasets, including two basketball datasets and three youth football datasets. All these datasets are acquired by a set of wide-angle high-resolution video cameras. The spatial relation between all cameras and the sports court is accurately calibrated based on the method mentioned above.

The sport courts are discretized into various numbers of grids due to various sizes of different courts in order to keep each grid have the same size as a typical player standing on the court. The different dataset has different parameters of cameras and sports video extracting. The characteristics of these datasets and the corresponding parameters can be seen in Table 3.1.

Note that the grid size should be a typical player size (for example, $0.5 \times 0.5m$)

Dataset	Camera	Resolution	fps	Frame	Players	Grids	Size(m)
STU	8	1280×720	24	8,000	10	112×60	0.4×0.4
STU0928	8	1920×1080	24	8,000	10	112×60	0.4×0.4
LH0716	8	2560×1920	25	10,000	10	140×76	0.5×0.5
LH0716v2	8	2560×1920	25	10,000	10	140×76	0.5×0.5
LH0928	8	1920×1080	20	10,000	16	140×76	0.5×0.5

Table 3.1 : The characteristics of our collected datasets and the corresponding parameters.

when the player is standing on the sports court. For different sports courts, the numbers of grids are generally different to keep the size equal.

The camera arrangement methods of these datasets are illustrated in Figure 3.1. An example of the STU basketball dataset is shown in Figure 3.3. An example of the LH0716v2 youth football dataset is shown in Figure 3.4.

3.2 Problem Formulation for 3D Localization

In this section, we intend to formulate the problem of 3D localization for multiple objects in a pre-defined sports field or space. This formulation will be widely applied in sports scenarios such as basketball, football, and volleyball.

In order to accurately estimate the 3D locations for multiple objects, we first define a 3D world coordinate system for the sports space, which is common to all sports players and has coincident content for all cameras. This coordinate system uses accurate camera calibration to associate objects' pixel coordinates with their likely 3D locations. Afterward, we define a set of discrete random variables to describe the status of targets' presence or absence on all discretized locations. Finally, we model the objects' occupancy probabilities as the posterior probabilities by ap-



Figure 3.3 : An example of the original image frames of our collected STU basketball dataset. Here shows four out of eight camera views. The camera arrangement method is illustrated in Figure 3.1.



Figure 3.4 : An example of the original image frames of our collected LH0716v2 youth football dataset. Here shows eight camera views. The camera arrangement method is illustrated in Figure 3.1.

plying the Bayesian algorithm under the given image sequences as the likelihood probabilities and a set of initialized values as the prior probabilities.

3.2.1 Modeling the sports space

We limit targets' moving area to a fixed 3D space, with perfectly calibrated cameras installed above the head height. These cameras capture multiple views of video streaming with overlapping fields based on a unified timestamp generator.

We let $\mathbf{C}, \mathbf{C} = \{1, 2, \dots, C\}$ to denote the index of cameras.

In order to accurately locate the 3D positions of these targets, we construct a 3D world coordinate system for the sports space. We define a ground plane for this 3D space with fixed size and orientation on where targets are standing. Based on this plane, we discretize the area common to all views into a certain number of square grids $\mathbf{G}, \mathbf{G} = \{1, 2, \dots, G\}$ with a fixed width \mathcal{W} . An example of these grids $k, k \in \mathbf{G}$ can be seen in Figure 3.5.

When a target is standing on the ground plane, we use one grid that has the minimum distance from it to describe the target's 3D coordinate. Suppose we add an average human height for the target. In that case, the grid when becomes a 3D cube with coordinate $(\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{Z})$, where $(\mathcal{X}, \mathcal{Y})$ denotes the center point coordinate of the grid, and \mathcal{Z} represents this average human height. An example of the 3D cube can be seen in Figure 3.6.

We let this cube visually represent a target's occupancy. We select a plane that is parallel above the ground plane to be the head plane, and its height equals the average human height we defined.

When we back-project the 3D cube to each camera view, this cube on the grid k can be seen as several rectangles in some of these camera views (We would have C rectangles if the cube is visible in all camera views). In each camera

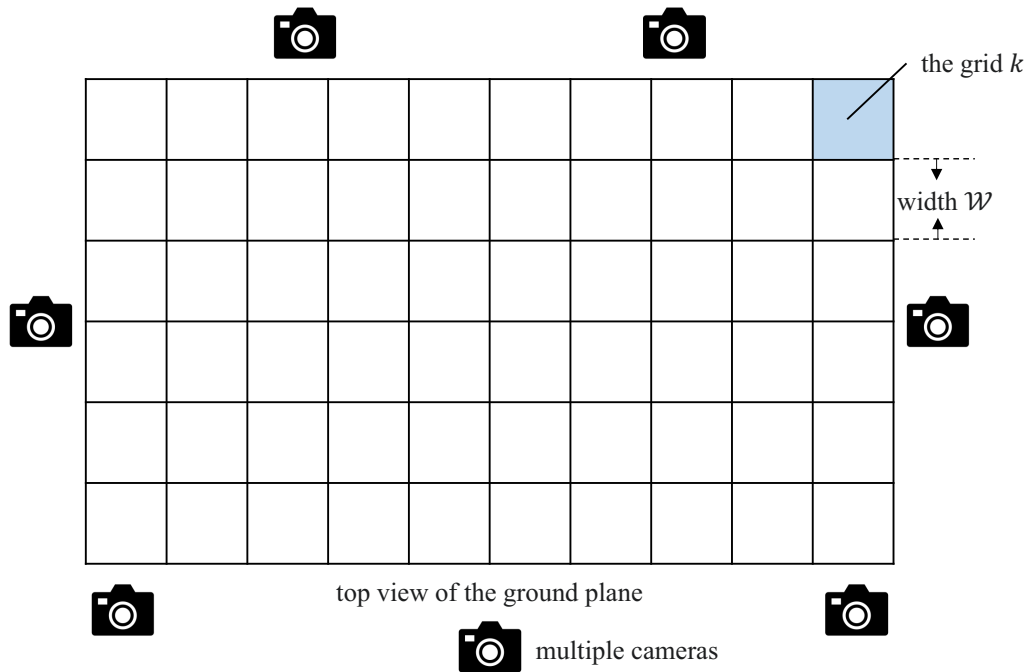


Figure 3.5 : Top view of the ground plane and the discretized grid cells, every square grid has the same width \mathcal{W} , the number of these grids is G . The number of cameras varies from different datasets.

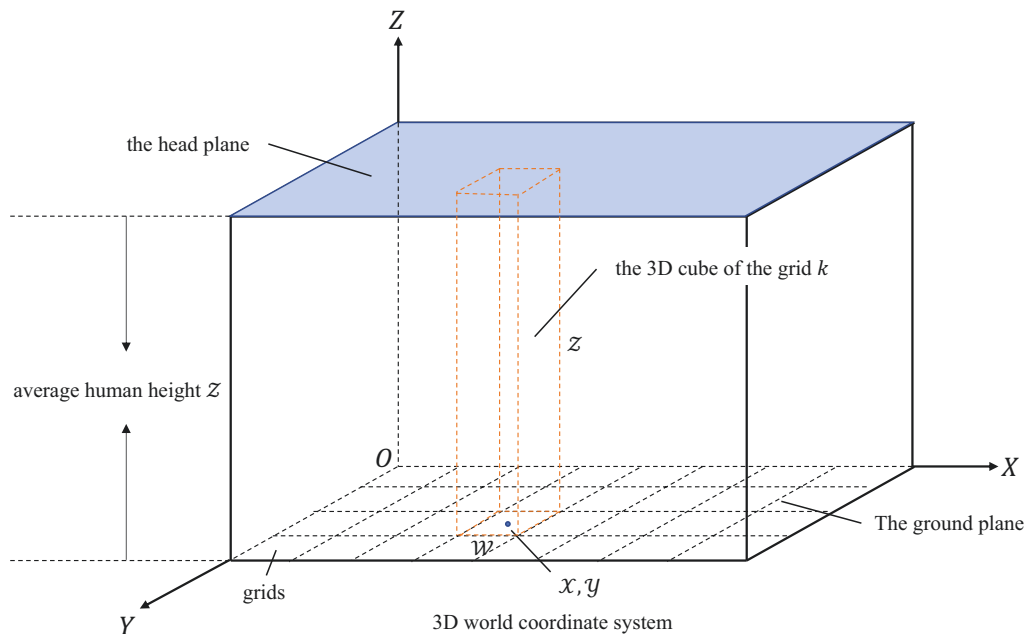


Figure 3.6 : The 3D world coordinate system with a cube. We use $(\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{Z})$ to denote the 3D coordinate of the cube.

view where the rectangle is visible, the rectangle has a 4-element 2D coordinate: $X_{min}, Y_{min}, X_{max}, Y_{max}$, which demonstrates the relation between the 3D cube and its corresponding 2D rectangle, as illustrated in Figure 3.7.

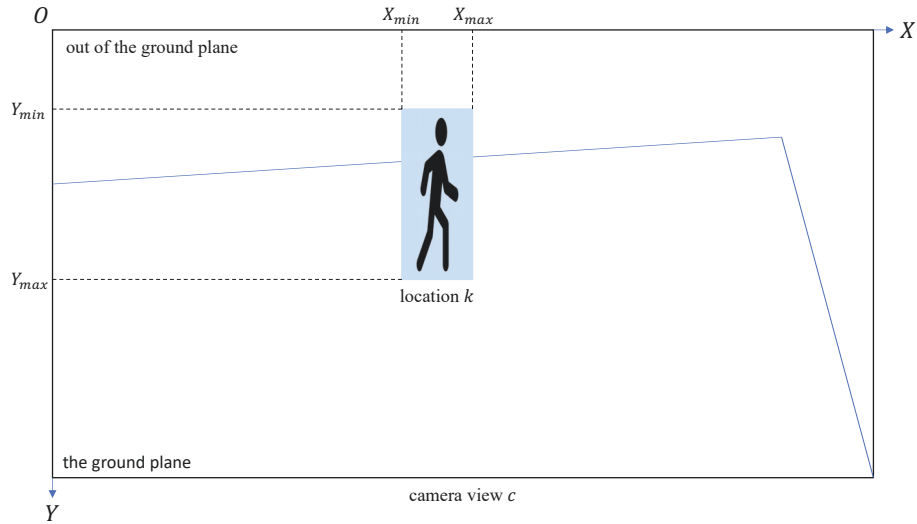


Figure 3.7 : Back-projection of the cube on location k in camera view c . The ground plane is shown inside the blue lines. The occupancy of a target is described as a rectangle at the corresponding location. Note that some cubes' back-projected rectangles may not be visible if the camera views don't include the field where the location belongs.

This 2D coordinate can be obtained based on accurate camera calibration. Notice that if a camera view cannot cover all the fields of the ground plane, some cubes may not be visible in that view, i.e., the corresponding rectangles are invisible in that view.

After multiple cameras film the video streaming, we extract single frames from all views based on the same timestamp t , yielding a set of image sequence $\mathbf{I}_t = \{I_t^1, I_t^2, \dots, I_t^C\}$. Based on these extracted image sequences, we aim to estimate the probabilities of all the targets with specific identification who are occupying those grids.

3.2.2 Statistical modeling for locations of sports objects

In order to model the status of presence or absence of a defined grid occupied by a sports object, we introduce a statistical model to describe this problem. If all the existing grids are modeled, we will obtain the status of multiple objects that we aim to estimate.

Firstly we denote a set of discrete random variable $\mathbf{X} = \{X_k | k \in \mathbf{G}\}$, where $X_k \in \{0, 1\}$. We let the Boolean random variable X_k represent the presence and absence of an individual standing on location k . $X_k = 1$ represents the presence, while $X_k = 0$ represents the absence. Thus, the probability of an individual standing on location k can be written as $P(X_k = 1)$. While absence of the location k is $P(X_k = 0)$.

Let $\mathbf{B} = \{B^1, B^2, \dots, B^C\}$ denote the information that is processed from the synchronized sport image sequences from all cameras \mathbf{C} , $\mathbf{C} = \{1, 2, \dots, C\}$, we then define the presence of an individual stand at location k as the conditional probability $P(X_k = 1 | \mathbf{B})$.

From common sense, we conclude that individuals in the sports space do not take into account the presence of other individuals in their vicinity when moving around. Additionally, all statistical dependencies between different views are due to the presence of individuals in the sports space. Thus this implies that as soon as the presence of all individuals is known, the views become independent.

As a result, by providing the prior probability $P(\mathbf{X})$, the likelihood probability $P(\mathbf{B} | \mathbf{X})$, we apply the Bayesian algorithms and obtain the posterior probability

$$P(\mathbf{X} | \mathbf{B}) = P(\mathbf{X})P(\mathbf{B} | \mathbf{X}) \quad (3.1)$$

Thus the posterior probability becomes tractable and easy to calculate, as long as we obtain the prior probability $P(\mathbf{X})$, and the likelihood probability $P(\mathbf{B} | \mathbf{X})$.

Chapter 4

POM+CNN+IniSet Localization Method

The Probabilistic Occupancy Map (POM) [42] algorithm has achieved good performance in 3D pedestrian detection and localization. Many other tracking-by-detecting methods were then developed based on this framework. Those approaches apply traditional background subtraction methods for multiple cameras and rely on accurate camera calibration to discretize the pre-defined ground plane into separate locations. They typically construct a simple Bayesian model for each location and estimate occupancy probabilities for these locations. However, those methods still rely on background subtraction as the most critical input, which is usually not discriminative enough to produce transparent foreground masks, especially when the background pixels keep varying, or the foreground density increases.

Furthermore, exact camera calibration results are significant to the detection performance. If cameras are inappropriately arranged, or the overlapping fields of view are not wide enough, false-positive detection would severely increase. Besides, the Bayesian models proposed in those methods tend to be increasingly complex, reducing the computational efficiency.

In order to tackle the problems mentioned above, in this chapter, we develop the POM+CNN+IniSet (POM+Convolutional Neural Network+Initialization Settings) localization method for sports videos. Since the ordinary background subtraction methods cannot provide discriminative foreground masks, we apply the CNN-based monocular object detection method jointly on multiple cameras to generate clear and correct foreground masks. We use those foreground masks containing players'

segmentation and coordinates to replace pixel-wise binary background subtraction. As a result, ambiguous and false-positive detection results caused by unclear foreground input are successfully eliminated. Moreover, we take advantage of two fisheye cameras arranged above the head of the sports court and develop a generic Bayesian model to initialize a set of indicative parameters. We use the foreground masks produced by the two fisheye cameras to pre-define those parameters with higher or lower initial values. This scheme can effectively avoid false-positive detection.

We divide the POM+CNN+IniSet method into two contributions and present the contributions in Section 4.1 and Section 4.2, respectively. We then conduct experiments and discuss the results in Section 4.3.

The first contribution of this method is that we apply a CNN-based monocular object detection method jointly on multiple cameras to generate clear and correct foreground masks. We use foreground masks containing players' segmentation and coordinates to replace pixel-wise binary background subtraction. The outcomes of our detection method consist of bounding boxes, segmentation masks, and pixel coordinates for each player. Compared with the traditional foreground subtraction methods that only provide pixel-wise binary foreground masks, our method enriches the input evidence of the 3D localization algorithms. We then add a refinement scheme to select detection results by removing undesired segmentation masks and creating a complete foreground image for each camera.

The second contribution of the method is that we take advantage of two fisheye cameras arranged above the head of the sports court and develop a generic Bayesian model to initialize a set of indicative parameters. We use the foreground masks produced by the two fisheye cameras to pre-define those parameters with higher or lower initial values. We found that from fisheye camera detection results, we can preliminarily define a certain number of grids that are most likely to be occupied.

The most important reason to use fisheye camera detection is that, from the fisheye cameras, the grids are more easily to be modeled according to the top view location settings (see Section 3.2). By inputting the prior probabilities of those pre-defined grids into the Bayesian model, we can improve the setting of prior probabilities and obtain more reliable posterior probabilities. The convergent speed is also improved.

4.1 2D Monocular Segmentation for Multiple Sports Players

For some typical sports datasets such as basketball and football, the foreground masks generated from multiple cameras produced by the conventional background extraction approaches usually have large-scale noises because of the motion of off-court people, the reflection of high-intensity headlights, and switching of billboards around the court. These results significantly reduce the accuracy of 2D detection and 3D localization.

Furthermore, those conventional approaches usually rely on pixel-wise foreground masks and multi-camera calibration as the only input. However, these pixel-wise foreground masks do not include any 2D locations, which can be obtained by 2D segmentation and significantly improve the 3D localization performance.

To address the problems mentioned above, we implemented 2D object detection and segmentation jointly on multi-camera image sequences to obtain clear and accurate foreground masks and their 2D bounding box information.

To do this, firstly, we implement Mask-RCNN [20] based monocular detection algorithms to process the original image frames from multiple cameras that are generated from the pre-processed sports videos. We then obtain sports players' 2D detection and segmentation results of each frame from each camera view. Secondly, from those detection results, we only select valid results classified as the "person" and ignore all the other classes. At the same time, we eliminate those detection

results which appear to be presented outside of the court. Finally, we use these selected 2D detection and segmentation results to generate transparent, accurate foreground masks for each frame and each camera. At the same time, all the empty areas are set to be the background. This procedure is illustrated in Figure 4.1.

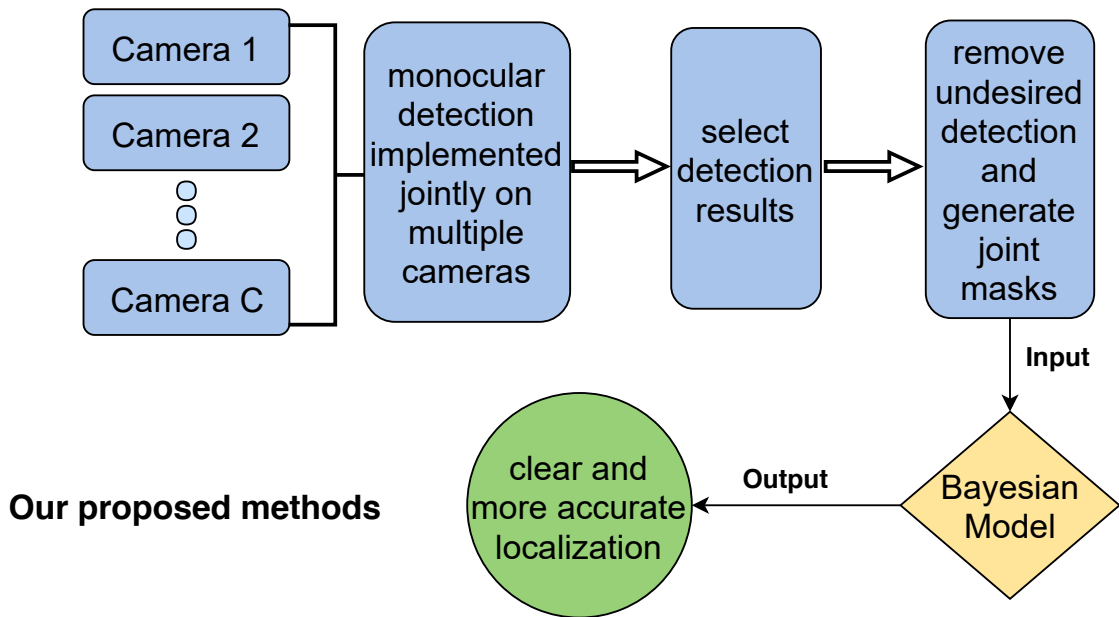


Figure 4.1 : This figure is an illustration of the implementation of 2D monocular segmentation for multiple sports players. We use CNN-based object segmentation to generate clear and correct foreground masks for the sports players that are captured by multi-camera. The outcomes of this method contain sports players' segmentation masks and bounding boxes. Those masks are removed when the players are standing outside of the sports courts.

This implementation removes the noises of background extraction and avoids missed foreground masks. Moreover, the outcomes of this implementation consist of detection results of every object, which are bounding box, segmentation mask, and pixel coordinate. Compared with the previous method, which only has pixels extracted from the background, we can enrich the inputs of the 3D localization system. After implementing CNN detection, the localization system selects detection

results by removing undesired segmentation and then generates a joint foreground input with transparent masks. This input was lately calculated by a Bayesian model to obtain clear and accurate localization results.

4.2 IniSet for The Bayesian Iteration

We found that from the fisheye detection results, we can preliminarily define a certain number of grids that are most likely to be occupied. The most important reason to use fisheye detection is that, from the fisheye camera, the grids are more easily to be modeled according to the top view location settings. By inputting the prior probabilities of those pre-defined grids into the Bayesian model, we can not only improve the setting of prior probabilities but also obtain more certain posterior probabilities. The convergent speed is also improved.

In the iteration process of the previous algorithms, a consistent initial value of estimated probabilities, i.e., the prior probabilities on all locations, need to be pre-set. The algorithms use this value to start the first step to compute a set of average synthetic images and then consequently compute the distance between these synthetic images and the original binary foreground masks.

Considering the 2D detection results, we can obtain accurate 2D detection from every single camera, including bounding boxes and foreground masks of each player. Thus, we can use this detection information to pre-set those probabilities for all locations mentioned above in order to improve 3D localization accuracy by eliminating false positive detection on empty locations and avoiding true-negative detection on occupied locations.

Furthermore, binary masks from the fisheye cameras of the implemented basketball dataset have not been used in this stage because of inaccurate detection results implemented on these fisheye cameras. However, this information can effectively re-

duce the impact of occlusions between players. Thus, we use the fisheye cameras to pre-set a set of initial probabilities of locations as the prior probabilities and implement experiments to evaluate the possibility of improvement. We take our proposed method mentioned above to generate the algorithm’s input data incorporating the initialized probabilities.

Firstly, we discretize the whole basketball court into a group of grids with a consistent amount. Each grid has a statistic value that presents the probability of the player’s presence on that grid. For input foreground image $\mathbf{I}_t = \{I_t^1, I_t^2, \dots, I_t^C\}$ and location $k \in \mathbf{G}$, $\mathbf{G} = \{1, 2, \dots, G\}$, a consistent value ranged from 0 to 1 (typically 0.01) is initialized to each prior probability q_k at the first step of the iteration process. With this initialized value, algorithms are designed to pursuit an optimal result in the following iteration.

By considering foreground masks from fisheye cameras, we identify each input image with a certain number of grids by locating the 3D coordinate of foreground blobs with those corresponding grids nearby. We mark these locations as $\{i, j, \dots, p\}$. That is, by processing foreground masks of the fisheye cameras, we use these foreground blobs to identify a number of locations $k \in \{i, j, \dots, p\}$ which are possibly occupied by players. For these locations ks , we initialize a considerably higher value (0.05, for example) to the corresponding q_k in the first step of the iteration process. At the same time, all the other q_k s where $k \notin \{i, j, \dots, p\}$ are set to be 0. With initialized prior probabilities, the average synthetic images in the first step of iteration are then computed.

In the following steps, the probability of each grid q_k is re-computed until an optimal solution is found. With these initialized probabilities containing prior knowledge of players, we not only accelerate the computation but also improve the accuracy of results, as can be seen in Figure 4.2.

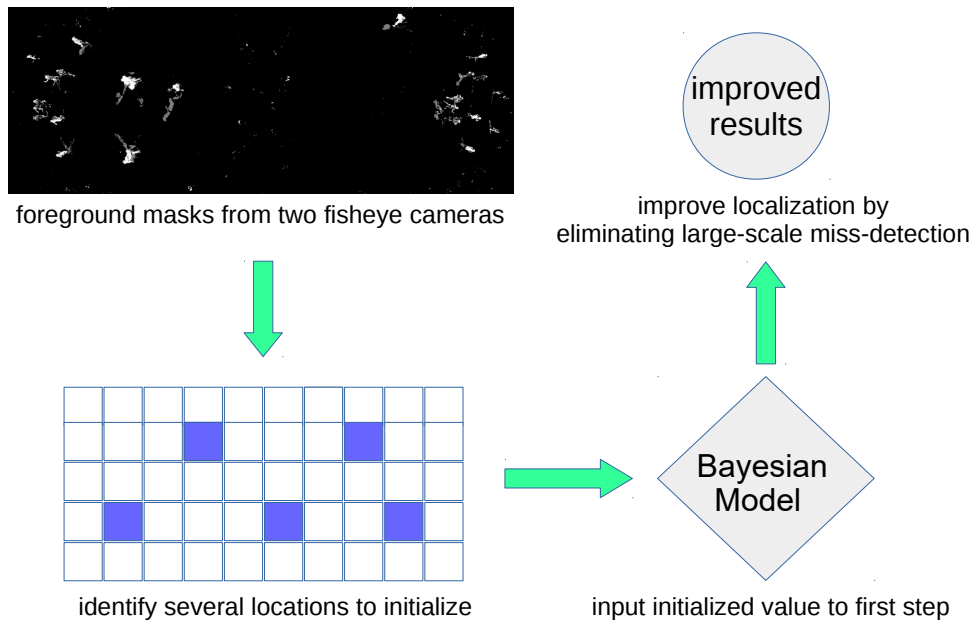


Figure 4.2 : Illustration of the IniSet model with indicative values, localization results are improved by eliminating large-scale miss-detection.

Note that this initialized value should typically be set considerably small to avoid false positive detection on absent locations. However, if being set extremely small, the algorithms may lose convergence, or take an extremely long time to get convergence in the iteration process, thus cause efficiency reduced.

4.3 Experimental Evaluation

In this section, we implement our proposed method POM+CNN+IniSet [34] and conduct experiments based on the APIDIS dataset. This dataset is a publicly available basketball dataset for multi-camera multi-target detection, localization, and tracking tasks. The experimental results are then compared with the baseline POM [42] based on the same dataset.

4.3.1 Datasets and metrics

We conduct our experiments on the APIDIS dataset. An example of the original image frame of this dataset is shown in Figure 4.3. This dataset is a publicly available basketball dataset for multi-camera multi-target detection, localization, and tracking tasks. This dataset includes five ordinary wide-angle cameras and two fisheye cameras. Those cameras are installed around the basketball court with various heights. In specific, the five ordinary cameras are set around 3 meters in height, while the other two fisheye cameras are installed considerably higher in order to obtain larger sights. The basketball court has the size of $2797 \times 1499\text{cm}$, being discretized into 128×72 totally 9216 grids. For each grid, the corresponding cube is designed to be $50 \times 50 \times 185\text{cm}$, with a 1.85m head plane. The basketball videos are acquired as 22 fps, 1600×1200 resolution. The five ordinary cameras are calibrated using the Bouguet Calibration Toolbox, while the other two fisheye cameras use the Kannala approach.



Figure 4.3 : An example of the original image frames of the APIDIS basketball dataset. Here shows seven camera views.

We give the experimental results as the probabilities of locations occupied by

players, which are very peaky. We therefore simply treat the location where the probability of presence is higher than 0.75 as a proposal. Comparing selected proposals with the given ground-truth locations, we use bird-eye view distance (BV) as the threshold to select the positive results.

Practically, as the APIDIS dataset discretizes the basketball court with many $500 \times 500mm$ grids on the ground plane, the distance between two neighboring locations is set to be $500mm$, which is a usual distance when two players are standing close. Consequently, we usually set the threshold BV to be $500mm$ to meet this criterion when selecting proposals. By applying the threshold with specific values, We count all produced proposals and obtain the number of correct proposals as true-positive (TP), missing proposals as false-negative (FN), and incorrect proposals as false-positive (FP). Note that we can apply various values to the threshold to obtain different sets of TP, FN, FP proposals. Given these, we can evaluate:

- Precision/Recall (P/R), which are taken to be $TP / (TP+FN)$ and $TP / (TP+FP)$, respectively.
- Multiple Object Detection Accuracy (MODA) [90], which will be provided as a function of the three selected thresholds. MODA assesses the accuracy aspect of system performance, and it utilizes the missed detection and false-positive counts:

$$MODA = 1 - \frac{\sum_{t=1}^{N_{frames}} (c_m(m_t) + c_f(fp_t))}{\sum_{t=1}^{N_{frames}} N_G^t} \quad (4.1)$$

Where the number of misses is indicated by m_t and the number of false positives is indicated by fp_t for each frame t , c_m and c_f are the cost functions for the missed detects and false positives and N_G^t is the number of groundtruth objects in the t th frame. c_m and c_f are used as scalar weights and can be varied based on the specific application. They were both equal (=1) in this evaluation.

- Multiple Object Detection Precision (MODP) [90], which uses the spatial overlap information between the ground-truth and the system output to compute the mapped overlap ratio:

$$MODP = \frac{\sum_{t=1}^{N_{frames}} \frac{Mapped\ Overlap\ Ratio}{N_{mapped}^{(t)}}}{N_{frames}} \quad (4.2)$$

Where the Mapped Overlap Ratio is:

$$Mapped\ Overlap\ Ratio = \sum_{i=1}^{N_{mapped}^{(t)}} \frac{|G_i^{(t)} \cap D_i^{(t)}|}{|G_i^{(t)} \cup D_i^{(t)}|} \quad (4.3)$$

Where $G_i^{(t)}$ denotes the i th groundtruth object in the t th frame, D_i^t denotes the detected object for G_i^t , and N_{mapped}^t is the number of mapped object pairs in frame t .

4.3.2 Results

Compared with POM that takes ordinary background subtraction as input, we implement 2D object segmentation to obtain accurate pixel-wise extraction and detection information jointly as input. Monocular segmentation is applied separately on the five ordinary cameras, while the segmentation results from each frame are jointly collected and then processed. To be specific, we implement the Mask-RCNN [20] approach to segment pixel-wise masks. The source code is based on Keras and Tensorflow, built on FPN and ResNet101 backbone. The model is trained by the MS COCO dataset.

After the processing of 2D segmentation, we then post-process the detection results. Firstly, only those players classified by class "person" are selected, and then their 2D coordinates are extracted. Afterward, abandon those players whose bounding box is out of the basketball court boundary. Finally, use remaining players to generate foreground masks, and at the same time, input their 2D coordinates.

As shown in Figure 4.4, compared with previous methods, unclear masks from the

ordinary background subtraction are eliminated, so are the consequently inaccurate localization results. Results produced by POM often have missed, duplicated, and complete false locations due to false foreground pixels. These issues are effectively solved by our proposed method.

We then identify a specific amount of regions where the players are likely to occupy. In order to connect the pixels of foreground masks with the encoding of those specific regions, calibration information of the two fisheye cameras is required. After those regions of interest are selected, we initialize a considerably small value to the probabilities affiliated with those corresponding grids in the iteration process. This process can generate a set of synthetic images which are more similar and closed with the original foreground masks. The examples of experimental results can be seen in detail in Figure 4.5. Note that the green lines present localization results with probabilities higher than 0.8, this kind of result is mostly accepted in the tracking process in future research. While the red lines describe probabilities range from 0.2 to 0.8, which are likely occupied by players but might be miss-detected or false-detected by algorithms.

As shown in period 1, frame 262, and frame 270, large-scale miss-detected results of the POM method are greatly improved. This issue appears commonly when the overlapping fields of view of those cameras are inadequate. While in period 3, frame 1810 and frame 1820, miss-detected results that appear in extremely crowded scenes are considerably improved. This example proves that our method reduces the impact caused by occlusion. In period 3, frame 1993, and period 4, frame 357, positive detection with low probabilities is refined to obtain satisfying outcomes; this reduces the inaccuracy and uncertainty of the POM localization methods. Thus, the performance of the POM approach is significantly improved by our proposed method.


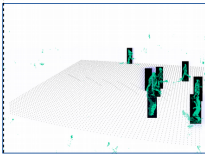


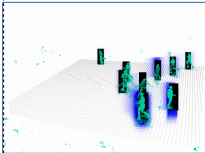

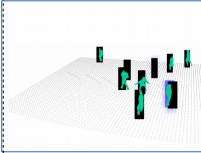

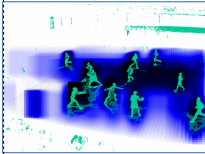

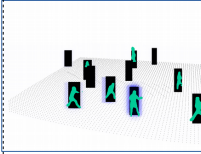

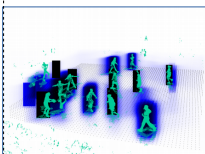

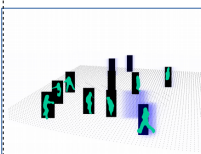

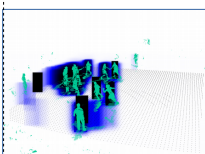

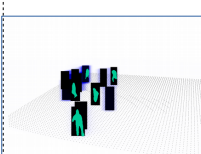
frame	traditional methods		our proposed method	
	background subtraction	localization results	2D monocular detection jointly applied to multi-cameras	localization results
	20			
40				
60				
90				
150				

Figure 4.4 : Examples of experimental results of our proposed method. The green masks show where the players are presented. The black rectangles represent localization with qualified probabilities, while the blue areas denote likely occupied locations.

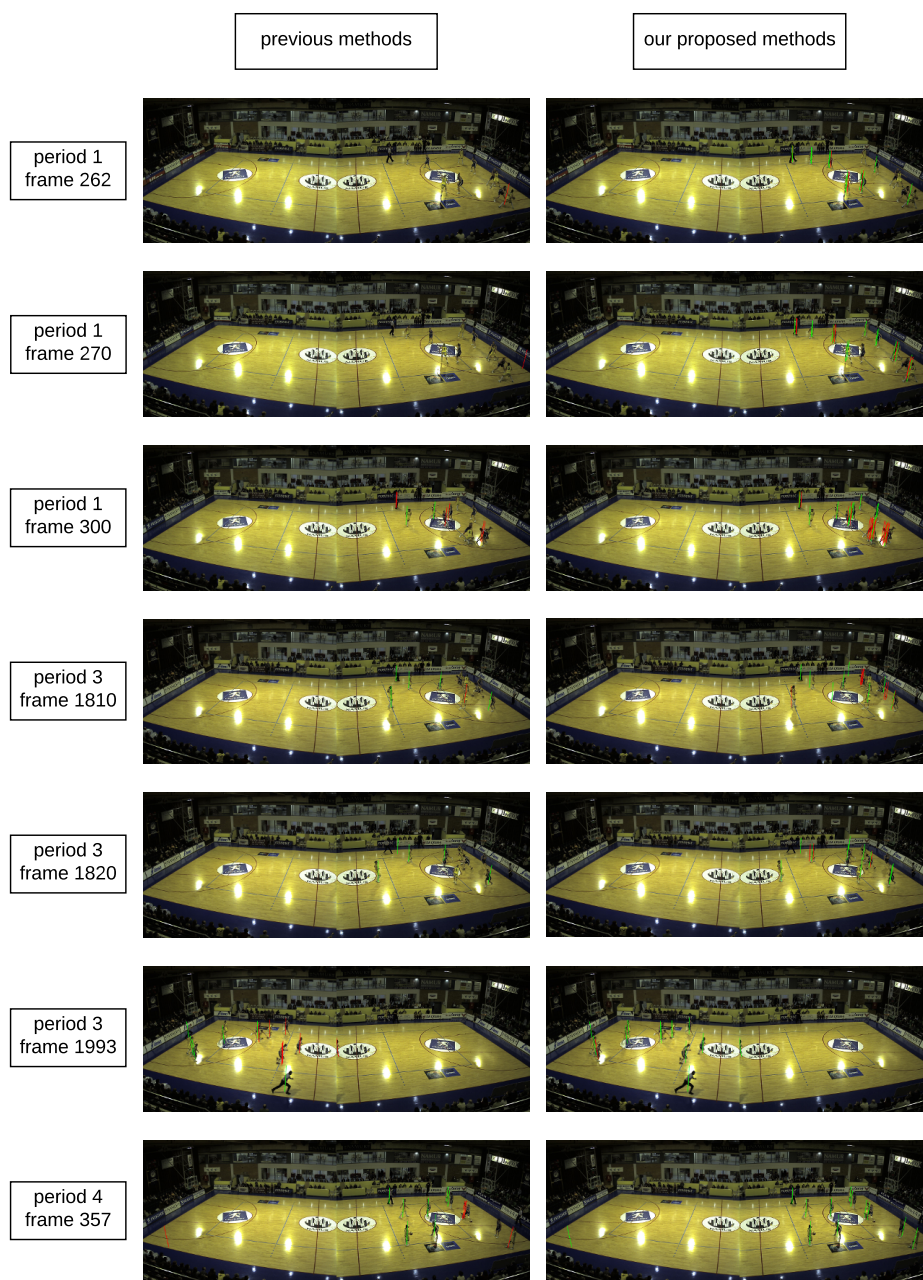


Figure 4.5 : Examples of experimental results of our proposed method, the left column is the implementation of the POM method, while the right side is ours. Different rows present for different frames selected from various periods. The green lines denote qualified localization with probabilities higher than 0.8, while the red lines represent locations that are less likely to be occupied, with probabilities ranging from 0.2 to 0.8.

Afterward, controlled experiments and quantified evaluation are implemented on the previous POM method and our proposed methods with ablation studies. The ground-truth data is obtained from the APIDIS dataset. The ground-truth data has 1000 synchronized frames taken from the seven cameras, with annotated locations of all the identified players in each frame.

Comparing with the ground-truth, we apply three sets of experiments on the POM method and our proposed method with two sets of ablation studies, which are POM+CNN and POM+CNN+IniSet. We analyze and select the localization results by three categories: True Positive (TP), False Positive (FP), and False Negative (FN).

Note that we identify a detection result as a positive detection only if the distance of its location from the ground-truth is less than $r = 0.5m$, and the selected threshold is 0.8. That is, we selected detected locations with probabilities higher than 0.8 as the true-positive detection results. The analysis of three experiments can be seen in detail in Table 4.1.

Methods	TP	FP	FN
POM	6770	900	4355
POM+CNN	8820	170	2305
POM+CNN+IniSet	9630	125	1495

Table 4.1 : The amount of TP, FP, FN detection results from three methods implemented on 1000-frame groundtruth data.

By using those data analyzed above, we use Precision, Recall, F-measure, the Multiple Object Detection Accuracy (MODA), and the Multiple Object Detection Precision (MODP) [90] as measurements to evaluate these multiple approaches. Statistical results can be seen in detail in Table 4.2. For Precision, Recall, and

F-1, we had significant improvements compared with the previous POM methods, especially when the occlusion plays a fundamental role in object detection.

Methods	Precision	Recall	F-1	MODA	MODP
POM	88.27%	60.85%	72.04%	52.76%	45.62%
POM+CNN	98.11%	79.28%	87.70%	77.75%	46.73%
POM+CNN+IniSet	98.72%	86.56%	92.24%	85.44%	47.04

Table 4.2 : Analysis results of the three control experiments. Precision/Recall and the F-1 for different methods when $r = 0.5m$, threshold is 0.8. The MODA and MODP are computed throughout the 1000-frame performance.

What is more, when the multiple cameras are not appropriately installed, the overlapping fields of view are inadequate for multiview targets, the previous POM method cannot achieve satisfying performance. In contrast, our proposed IniSet scheme reaches better results. As for the MODA and MODP, compared with the POM method, our proposed approaches have considerably better performance. Compared with the POM+CNN scheme, the POM+CNN+IniSet even achieves better performance under those metrics.

Also, we select various threshold BV (ranging from 0.1 to 0.99) to evaluate the Precision and Recall of those three schemes and present the Precision-Recall curve shown in Figure 4.6. As can be seen, our proposed methods have considerably higher Precision and Recall than POM, and the P-R curve shows that our methods are more robust than POM.

4.4 Conclusion

In this chapter, we proposed the POM+CNN+IniSet 3D localization method. This approach applies the CNN-based monocular object detection method jointly on

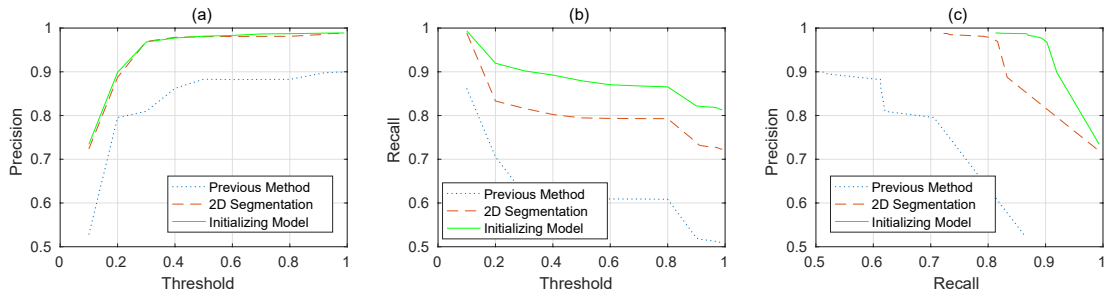


Figure 4.6 : Precision and Recall curves in terms of the different thresholds. The first column is Precision versus different thresholds of three methods. The second column demonstrates Recall various from different thresholds. Additionally the third column is the Precision-Recall curves of these three methods.

multiple cameras to generate clear and correct foreground masks. We use those foreground masks containing players' segmentation and coordinates to replace pixel-wise binary background subtraction. As a result, ambiguous and false-positive detection results caused by unclear foreground input are successfully eliminated. Moreover, we take advantage of two fisheye cameras arranged above the head of the sports court and develop a generic Bayesian model to initialize a set of indicative parameters. We use the foreground masks produced by the two fisheye cameras to pre-define those parameters with higher or lower initial values.

We then implement our proposed method POM+CNN+IniSet and conduct experiments based on the APIDIS dataset. The experimental results are then compared with the baseline POM based on the same dataset. Experimental evaluation demonstrates that our proposed method outperforms the baseline POM by a large margin. This method enriches the localization system's input information, removes undesired segmentation masks, improves the precision and recall of the localization outcomes, and boosts the localization performance.

Chapter 5

PomID Localization Method

POM+CNN+IniSet localization method proposed in Chapter 4 can eliminate ambiguous and false detection results caused by unclear foreground input, effectively avoid false-positive location results, improve the localization performance. We apply a CNN-based monocular object detection method jointly on multiple cameras to generate clear and correct foreground masks and take advantage of the two installed fisheye cameras to initialize a set of indicative parameters for the Bayesian inference model.

However, after a large amount of literature review, we concluded that traditional localization methods, which do not consider the targets' identities, also known as the non-ID localization methods, have significant drawbacks for the localization performance and multi-object tracking tasks. These drawbacks include heavy occlusion in extremely crowded sports scenes, ambiguous extraction input, tiny pixel blobs in remote views, and rapid-moving targets in sports games.

Most importantly, these non-ID localization approaches cannot be associated and retrieved back to the particular objects. Their localization results are not distinguishable and not unique, thus would cause more false-positive results. Additionally, these methods cannot solve the identity switches when multiple players are standing close and moving around.

In order to overcome these drawbacks, we develop the PomID (POM Identity) method in this Chapter and afterward the PIOM (Probabilistic and Identified Occupancy Map) method in Chapter 6. The PomID method applies a DeepPlayer

model, including a Cascade Mask-RCNN model and a pose-guided partial feature embedding to conduct object segmentation and identification for multiple sports objects. The DeepPlayer model produces both the individuals' foreground masks and their identities, which are treated as the given evidence for the 3D localization algorithms. This method then separately estimates the likely location for each player who has certain and correct identity input and jointly calculates the results for the rest targets without ID labels. Final outcomes are then refined by a set of reasonable constraints. The PomID method includes multiple objects' identities as evidence to estimate the likely occupied locations, making the localization results distinguishable and unique to be associated with the particular objects. This method can accurately locate multiple objects and effectively avoid identity switches for multiple-object detection and tracking tasks.

The PomID localization method consists of two parts. Firstly we develop a DeepPlayer model including a Cascade Mask-RCNN model and a pose-guided partial feature embedding to conduct object segmentation and identification for multiple sports players. Afterward, we take outcomes of the DeepPlayer model as the given evidence for the PomID localization algorithm, which separately estimates the likely location for each player who has a confident identity and processes the other players together whose identities are uncertain. Finally, we apply a refinement scheme to obtain optimal results.

In Section 5.1, we introduce the DeepPlayer model for multi-object segmentation and identification. Then we illustrate the PomID localization scheme in Section 5.2. Experimental evaluation is presented in Section 5.3.

5.1 DeepPlayer Model

The DeepPlayer model is proposed to obtain each player's identity. This model contains two parts:

1. the Cascade Mask-RCNN for coarse-grained player segmentation and fine-grained jersey number recognition
2. the player segmentation embedding into a deep representation through posed-guided partial feature embedding (PoseID)

As shown in Figure 5.1, the Cascade Mask-RCNN model firstly detects each player and classifies the player by the team, and then segments the player's instance mask. Then the model recognizes the jersey number from the detected player bounding box. In a nutshell, we obtain the team class and number class of the detected player if the jersey number can be detected. Otherwise, we extract the deep representation of the detected player by PoseID. Finally, we combine the jersey number class, the team class, and the pose-guided partial feature embedding to infer the player ID after fully connected layers.

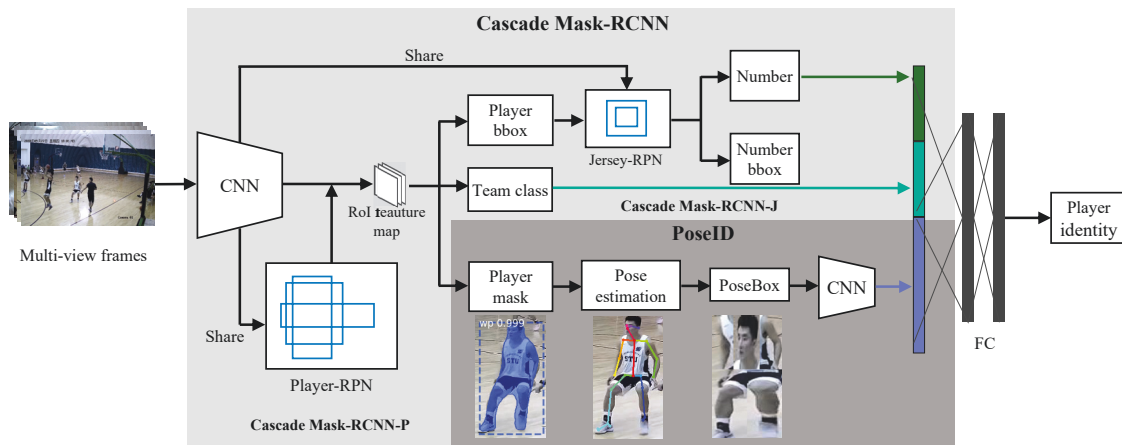


Figure 5.1 : The architecture of the DeepPlayer model. This model consists of two parts: (1) the Cascade Mask-RCNN for coarse-grained player detection(Cascade Mask-RCNN-P) and fine-grained jersey number recognition(Cascade Mask-RCNN-J); (2) the player mask embedding into the deep representation using PoseID. Finally, the player identity is decided by the jersey number class, the team class, and the deep representation.

5.1.1 Cascade Mask-RCNN

In terms of a player with a readable jersey number, we formulate player identification as player jersey number detection and classification since the jersey number/class can provide unique ID information. If a rough detector is employed to detect players and jersey numbers directly, it will produce inaccurate region proposals and miss-association of player and jersey numbers.

Therefore, we extend and modify the Mask-RCNN [20], which is a CNN-based detector for detection and instance segmentation. We propose a Cascade Mask-RCNN model, which includes two parts: (1) Cascade Mask-RCNN-P for player detection and instance segmentation under coarse granularity, (2) Cascade Mask-RCNN-J for jersey number detection and recognition under fine granularity.

Firstly, we detect all players from multiview images to obtain the bounding box, the team categories, and the instance segmentation of each player. Then, the player bounding boxes are put into a jersey number localization model to detect number location, followed by a number classification model to recognize the jersey number. To reduce the duplicate calculation, both RPN of the player and RPN of the jersey number share the CNN feature map of the input image. Finally, we save the team class (a 3-dimensional vector) and the jersey number (a 24-dimensional vector) for subsequent processes.

We leverage a ResNet-50 [18] model to extract CNN feature of input frames and share the feature to Player region proposal network (P-RPN) and Jersey number region proposal network (J-RPN), to generate Region of Interest (RoI) feature map of the player by Anchor. Then, the player bounding box (bbox) is predicted by regression, and the team class is predicted by classification. The player mask is a one-hot $m*m$ binary key-point where the pixels that belong to the mask are labeled as foreground. The team class contains two teams and a referee. The background

and audience are defined as background. $\mathcal{L}_{cls}(p_i^c, g_i^c)$ is the loss of team classification, and $g_i^c \cdot \mathcal{L}_{loc}(p_i^l, g_i^l)$ is the loss of player bounding box regression, and $\mathcal{L}_{mask}(p_i^m, g_i^m)$ is the loss of player mask. The loss of player \mathcal{L}_{ply} is defined as:

$$\mathcal{L}_{ply} = \sum_i \mathcal{L}_{cls}(p_i^c, g_i^c) + \sum_i g_i^c \cdot \mathcal{L}_{loc}(p_i^l, g_i^l) + \sum_i \mathcal{L}_{mask}(p_i^m, g_i^m) \quad (5.1)$$

Where g_i^c and p_i^c indicate ground-truth and predicted classification of the proposal region. p_i^l is the predicted vector representing the offset between the i th proposal and its corresponding ground-truth bounding box, and g_i^l is the true offset value between them. g_i^m and p_i^m represent ground-truth and predicted mask of the proposal region. We use Softmax as the loss function of \mathcal{L}_{cls} and SmoothL1 as the loss function of \mathcal{L}_{loc} , respectively. Compared with Euclidean distance, SmoothL1 can reduce the outlier effect and make the model converge faster.

After the player detection, the J-RPN calculates the jersey number bbox from the detected player bbox, and classifies the jersey number bbox. In this work, we treat jersey number recognition as a detection problem. We model all occurring jersey numbers as a separate class. In this case, this is a 24-class classification problem, as not all numbers appear in the dataset. For positive samples, there is a restriction that the jersey number bounding box must be included in the responding player bounding box. The loss of jersey number \mathcal{L}_{jrs} is defined as:

$$\mathcal{L}_{jrs} = \sum_i \mathcal{L}_{cls}(j_i^c, h_i^c) + \sum_i h_i^c \cdot \mathcal{L}_{loc}(j_i^l, h_i^l) \quad (5.2)$$

Where h_i^c and j_i^c indicate ground-truth and predicted the classification of the proposal region. j_i^l and h_i^l are the predicted offset and true offset values between the i th proposal region and its corresponding ground-truth bounding box.

Our full objective for an image is defined as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{ply} + \lambda_2 \mathcal{L}_{jrs} \quad (5.3)$$

The hyper-parameter λ_1 and λ_2 control the balance between the two losses. Note that we train the P-RPN module only on fully labeled data, while we train the J-RPN module on both weakly labeled data and fully labeled data. The weak labels are player bounding boxes, and the full labels are jersey number bounding boxes. This kind of weakly supervised learning [84] can improve the efficiency of the network training compared to supervised learning.

5.1.2 Pose-guided partial feature embedding

Besides the jersey number and team class, we develop a posed-guided partial feature to represent a specific player to assist player identification.

To find the regions that distinguish a player from his teammates, we implement GRAD-CAM [91] on the player bounding box classification. We directly train an Inception V4 model, and the class is the identity of the player. We compute the gradient of the class output value with respect to the feature map. Then, we weigh the output feature map with the computed gradient values and average the weighed feature map along the channel dimension resulting in a heat map. Through observing the heat map, we reach a conclusion similar to [88]. Discernible details always appear in similar positions for each player. Similarly, the head part, sleeves, socks, and shoes look distinctive to the players. This is interpretable. Therefore, we propose the pose-guided partial feature embedding (PoseID) for player identification.

Player occlusion may cause multiple players to appear in one bounding box detected by the body key-points detector. This may lead to incorrect pose estimation owing to the player bounding boxes with impurities. Different from others' pose estimation by using detected player bounding box [88, 92], we localize the key-points from the pure player mask generated by our Cascade Mask-RCNN. This will avoid the incorrect pose estimation because the player mask is instance segmentation, which contains only one object. We adopt the off-the-shelf model of OpenPose [93],

which is an effective tool to detect the 2D pose of people in an image. We leverage 25-key-point body/foot key-point estimation. A set of 25 body joints are detected, i.e., face, neck, left and right shoulders, left and right elbows, left and right wrists, left and right hips, left and right knees, left and right ankles, and left and right feet, as shown in Figure 5.2.

According to the aforementioned player mask and pose estimation, we build a set of PoseBox, as shown in Figure 5.2. The PoseBox can eliminate background noise and correct the pose variations.

- PoseBox 1. This type is designed by Zheng et al. [92]. It includes the torso, two arms, and two legs. An arm consists of the upper and lower arms. A leg is comprised of the upper and the lower leg submodules.
- PoseBox 2. On the basis of PoseBox 1, we add the face. In our experiment, we show that PoseBox 2 is superior to PoseBox 1, thanks to the enriched information brought by the face.
- PoseBox 3. Based on PoseBox 2, we put subtract the torso box. We find that the subtraction of the torso brings performance increase. In our case, this increase is explicable because of the same jersey color.

After constructing the PoseBox, we adopt the ResNet-50 to extract the convolutional feature and then flatten it into a 2048-dimensional vector.

5.1.3 Obtain players' identification

To the end, we obtain the team class, jersey number class, and pose-guided partial feature embedding. The team class can be described as a 3-dimensional vector \mathbf{z}_1 containing each class with its probability. The jersey number class is a 24-dimensional vector \mathbf{z}_2 containing each class with its probability. The pose-guided

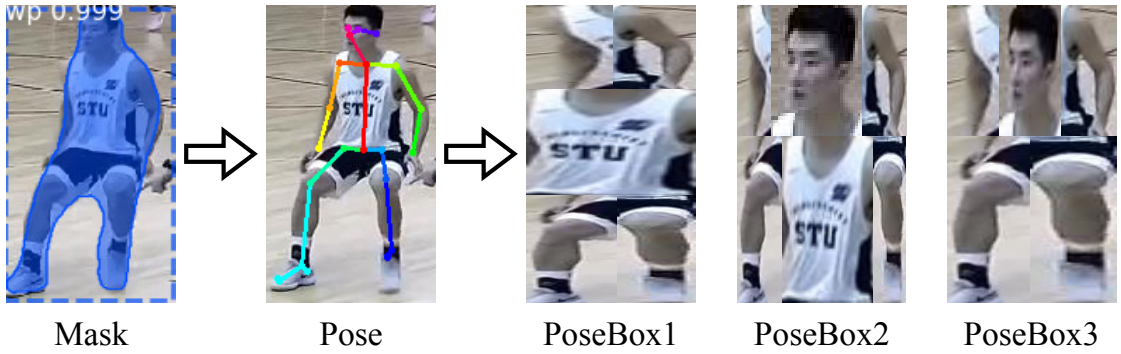


Figure 5.2 : PoseBox construction. Given a mask, the player pose is estimated by OpenPose. PoseBox1 = torso + arms + legs; PoseBox2 = head + torso + arms + legs; PoseBox3 = head + arms + legs.

partial feature can be described as a 2048-dimensional vector \mathbf{z}_3 , an embedding of the PoseBox.

We combine these three vectors as the input and construct a Softmax classifier with two fully connected layers to predict player identity. Since the confidence of the three vectors is different, the input vector is defined below:

$$\mathbf{z} = \mu_1 \mathbf{z}_1 + \mu_2 \mathbf{z}_2 + \mu_3 \mathbf{z}_3 \quad (5.4)$$

Where μ_1 , μ_2 , μ_3 control the weights of team class \mathbf{z}_1 , jersey number class \mathbf{z}_2 , pose-guided partial feature \mathbf{z}_3 , respectively. In our case, we set $\mu_1 = 1$, $\mu_2 = 0.5$, $\mu_3 = 0.25$, as the error increases progressively.

5.2 PomID Localization Scheme

In order to provide players' 3D locations with identities, we develop an effective model, PomID, by applying object segmentation and identification jointly on multiple cameras. This model not only provides objects' 3D locations but also gives distinguishable ID information for every target. This kind of localization result can be associated and retrieved back to the particular object that is believed to be

presented at the specific location.

As the aforementioned DeepPlayer model predicts the probabilities of the players' identities, some identities might be ambiguous when the estimated probabilities are considerably low. At the same time, the identities with high confidence of results have distinctive identification evidence. Thus, we develop the PomID model using two different strategies to process the identity estimation from the DeepPlayer model. The PomID model consists of two parts: (1) PomID localization scheme with identified ID labels for the individuals with high confidence of estimation; (2) PomID localization scheme with ambiguous ID labels for the individuals with ambiguous estimation results.

An overview of the PomID model can be seen in Figure 5.3. We develop the PomID model by taking the outcomes of the DeepPlayer model as input.

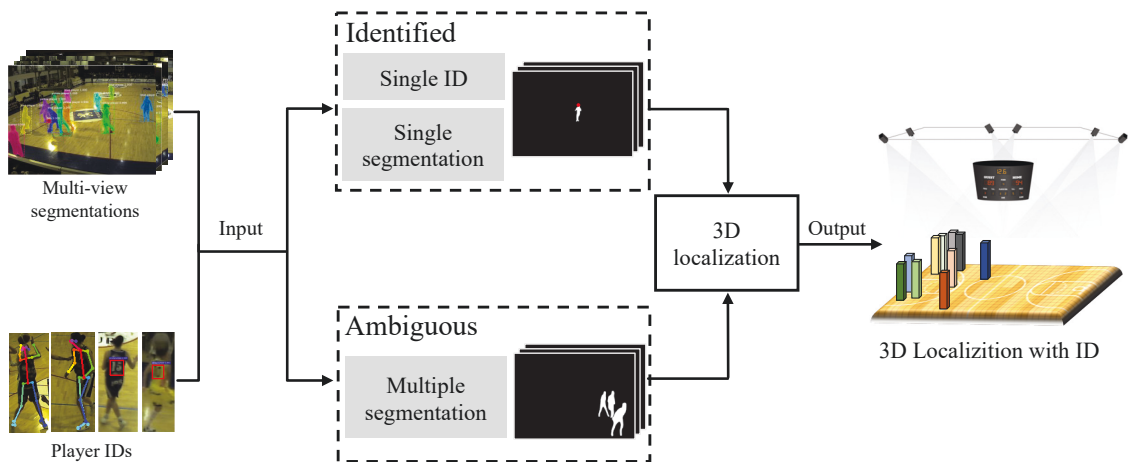


Figure 5.3 : Overview of the PomID model. The input of the PomID model includes objects' segmentation masks and ID labels. The 3D localization algorithm processes the players with identified ID and the players with ambiguous ID, respectively, followed by a post-process procedure with a set of experimentally defined thresholds. The output is the final 3D locations with distinguished identities.

The players' segmentation masks can be denoted as $\mathbf{B}_i|c \in \mathbf{C} - \mathbf{C}^{am}$, while i

indicates the identification label, \mathbf{C} denotes multiple cameras, and \mathbf{C}^{am} represents the camera where the identity estimation is ambiguous for ID i .

By inputting these segmentation masks \mathbf{B}_i , with identification i , we implement the 3D localization algorithm to calculate the probability of player i standing at location k , which can be presented as $P_k(X, Y, Z, i)$, where (X, Y, Z) denotes the player’s 3D coordinate, i denotes the player’s identity. Note that we use the discretized ground plane for the definition of the 3D world coordinate system. Thus the 3D coordinate Z is typically set to be 0.

In the case of the extreme occlusion, the accuracy of players’ identification estimation, especially for the occluded players, may be unavailable in some camera views $c \in \mathbf{C}^{am}$. For those players whose estimated identities are ambiguous, we use $\mathbf{B}_{am} = \{B_{am}^1, B_{am}^2, \dots, B_{am}^c\}$ to represent their segmentation masks and identification estimation. We then implement the 3D localization algorithm based on that input information to extract the probability of occupancy of the player with ambiguous identity on location k , which is $P_k(X, Y, Z, am)$.

Finally, we post-process those probabilities with ID by setting a threshold (experimentally 0.85) to extract a certain number of occupancy probabilities. To the end, the PomID model outputs the 3D locations and players’ ID labels.

5.3 Experimental Evaluation

In this section, we implement our proposed method PomID [35] and conduct experiments based on the publicly available dataset APIDIS and one of our collected basketball datasets STU. Then we analyze the performance of PomID based on those two datasets.

5.3.1 Datasets and experimental configuration

The information about the publicly available dataset APIDIS can be found in Section 4.3. Here we introduce one of our collected basketball datasets STU. This dataset is a university basketball match dataset we collected at Shantou University. It is a temporal synchronization dataset with eight wide-angle cameras. The video files are recorded at 24 fps in 1280×720 resolution in the format of MPEG-4. We have implemented our experiments on five periods of image sequences, which include almost 8,000 image frames. The basketball court is $28 \times 15m$. There are 16 players on the court, including two referees and two eight-player teams.

We use two different sets of experimental settings for those two datasets, respectively.

For the APIDIS dataset, we discretize the basketball court into rectangle grid cells with a size of 128×72 , each of which is named as location k (from index 0 to 9215). For each grid cell, the corresponding 3D cube is designed to be $50 \times 50 \times 185cm$, with the head plane as $1.85m$ height.

For the STU dataset, we discretize the basketball court into rectangle grid cells with a size of 112×60 , each of which is named as location k (from index 0 to 6719). For each grid, the corresponding 3D cube is designed to be $40 \times 40 \times 185cm$, with the head plane as $1.85m$ height.

5.3.2 Results

We firstly implement the PomID 3D localization method based on the APIDIS dataset. As shown in Figure 5.4, we back-project the obtained localization results onto the original image frames. Different colors indicate different team classes, while the labels of numbers indicate the different identities of multiple players across all the frames.

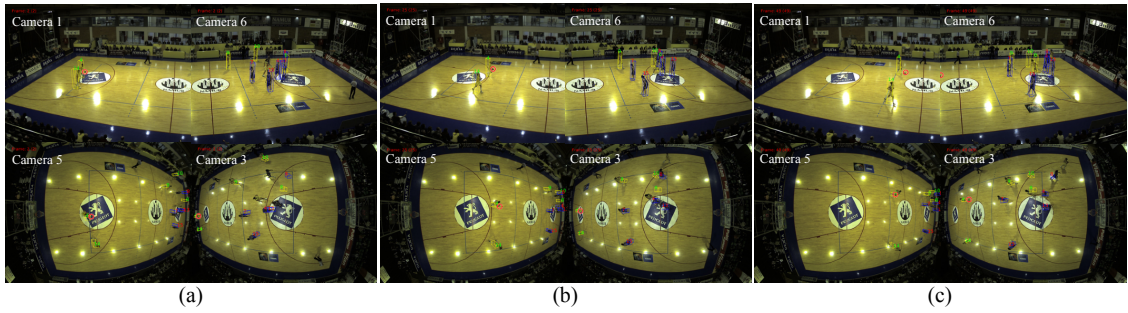


Figure 5.4 : Illustration of the PomID 3D localization results. The results are back-projected onto the original image sequences. Different colors indicate different team classes, while the labels of numbers indicate the different identities of multiple players across frames.

As can be seen in Figure 5.4, 3D localization results are accurately back-projected into original images from not only those ordinary cameras but also two fisheye cameras. Each estimated location has a unique ID label, which makes the localization results distinguishable and associated with the corresponding players. Thus, with this kind of localization results, we would know which player is believed to be presented on a known location and which location is proved to be occupied by a particular player. 3D localization results with exact ID labels across all the frames can effectively eliminate false-positive and false-negative locations and avoid duplicated locations.

Additionally, as the two referees are not considered into the problems of multiple players localization, we can use the identification label to ignore the useless estimated locations. This process is significantly essential for the sports video scenarios because it is considerably common that players who are out of the sports court stand close to the court boundaries.

As shown in Figure 5.5, we highlight the extremely crowded position both in ordinary view and fisheye view, inside the green boundaries. From the figure, we can

see that in the extremely crowded position, the PomID localization can effectively avoid identity switches among player 11, player 14, and player 15. Identity switches are widely existing in crowded sports scenes, which brings more challenges to the multiple object localization and tracking tasks.

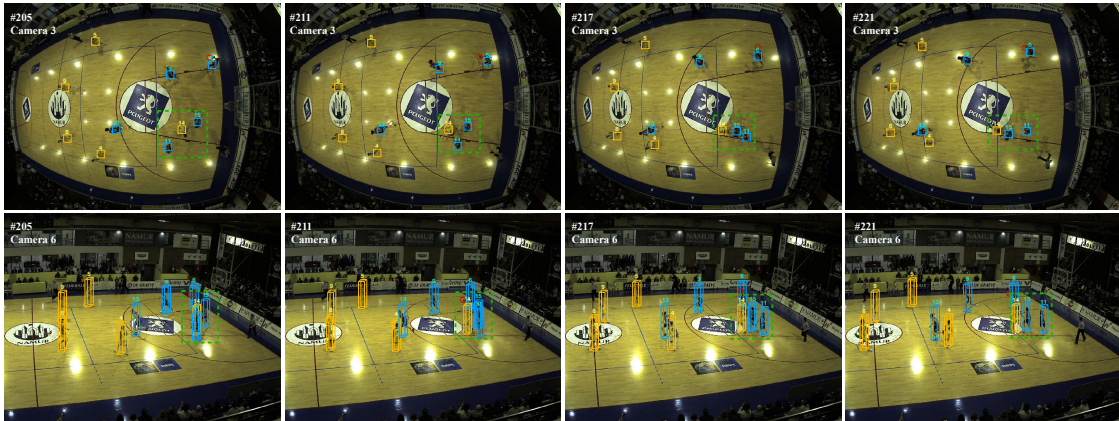


Figure 5.5 : Illustration of the APIDIS dataset in Camera 3 and 6 indicates that the proposed method can avoid identity switches among Player 11, Player 14, and Player 15 in the dashed box.

3D localization results with identified labels are unique among all the objects in one single frame and can keep the consistency across continuous image frames. Thus, our proposed PomID localization framework can not only obtain accurate localization results with unique identities but also avoid identity switches for further multiple object tracking tasks.

We then implement our proposed method based on the STU dataset. The details of the STU dataset are illustrated above.

We back-project the localization results with identity labels and illustrate the problems of identity switches, as shown in Figure 5.6. The red and yellow dotted bounding boxes are the back-projection of the locations estimated by the PomID algorithm, and the digital numbers upon the bounding boxes are the ID labels for

each player on the basketball court. Red and yellow, as well as green and light blue, indicate different team classes.



Figure 5.6 : Illustration of our localization results on the STU dataset in Camera 1, 4, and 6. The blue dotted boxes show that our method avoids identity switch between Player 6 of the white team and Player 11 of the black team.

Compared with the APIDIS dataset that uses higher installed cameras, the basketball court in the STU dataset is considerably smaller, and the basketball players are moving faster. Thus, occlusion tends to be more severe. This illustration indicates that our proposed PomID localization framework can also obtain good performance in the STU basketball dataset.

5.4 Conclusion

In this chapter, we proposed the PomID 3D localization method. This approach applies a DeepPlayer model including a Cascade Mask-RCNN model and a pose-guided partial feature embedding to conduct object segmentation and identification

for multiple sports objects. The DeepPlayer model produces both the individuals' foreground masks and their identities, which are treated as the given evidence for the 3D localization algorithms. This method then separately estimates the likely location for each player who has certain and correct identity input and jointly calculates the results for the rest targets without ID labels. Final outcomes are then refined by a set of reasonable constraints. The PomID method includes multiple objects' identities as evidence to estimate the likely occupied locations, making the localization results distinguishable and unique to be associated with the particular objects.

We then implement our proposed method PomID and conduct experiments based on the publicly available dataset APIDIS and one of our collected basketball datasets STU. Then we analyze the performance of PomID based on those two datasets. Experimental evaluation demonstrates that our proposed method can accurately locate multiple objects and effectively avoid identity switches for multiple-object detection and tracking tasks.

Chapter 6

PIOM Localization Method

The PomID localization method proposed in Chapter 5 jointly applies CNN-based object segmentation and object identification for multiple sports players, develops a Cascade Mask-RCNN model and a pose-guided partial feature embedding to obtain individuals' foreground masks and their identities. This method separately estimates the likely location for each player who has certain and correct identity input and then jointly calculates the results for the rest targets without ID labels. Final outcomes are then refined by a set of reasonable constraints.

However, the PomID method also has lots of drawbacks.

Firstly, its performance is extremely sensitive to the quality of players' identification. It is quite challenging to obtain all targets' identities. Once a target's identity is confirmed ambiguous, the localization algorithm cannot take advantage of the identification input of this target. Thus it is more likely to cause false-positive or true-negative results, reducing the system's performance. Afterward, it has a high volume of data to be pre-processed. Every target's segmentation and identification input from each image frame needs to be pre-processed. Thus it is very time-consuming to process enormous amounts of data. Finally, it has high demanding localization process that requires high CPU and GPU capacity. The localization process iterates as many times as the number of targets in multiple views from image sequences. It is extremely time-consuming to implement the localization algorithm on sports video scenarios because sports videos often contain a long period of time and over ten players to be detected.

Therefore we develop the PIOM 3D localization framework. The PIOM 3D localization method mainly contains three parts:

1. the Image&ID model and image distance norm for visually associating the image pixels with the occupancy and identification probabilities, calculating the errors between our synthetic images with ID modules and the given evidence of pixel-level segmentation and identification.
2. the estimation of the posterior probabilities and computation of the loss function.
3. the iterative process and convergent computation for the optimal solutions.

An overview of this framework is shown in Figure 6.1.

Firstly, we use the DeepPlayer model (see Section 5.1) that consists of a Cascade Mask-RCNN model and a PoseID model to extract the sports players' segmentation masks and identification labels at pixel-level.

At the same time, with the 3D world coordinate settings for the sports space mentioned in Section 3.2, we introduce an Image&ID model and an image distance norm to fuse the multiview pixel-wise segmentation and ID labels together with their 3D spatial relations. The Image&ID model consists of a set of synthetic images and synthetic ID modules. The synthetic images link the occupancy probabilities with the visible and computable image pixels, while the synthetic ID modules associate the identification inputs from all camera views with accurate spatial coordinates.

Afterward, we develop a multi-dimensional Bayesian model and then construct a loss function as the $K - L$ divergence between an estimated probability distribution and the true posterior probability. The prior probabilities are initialized at the beginning of the iteration, while the likelihood probabilities are approximated by the normalized image distances between the synthetic average images with ID modules

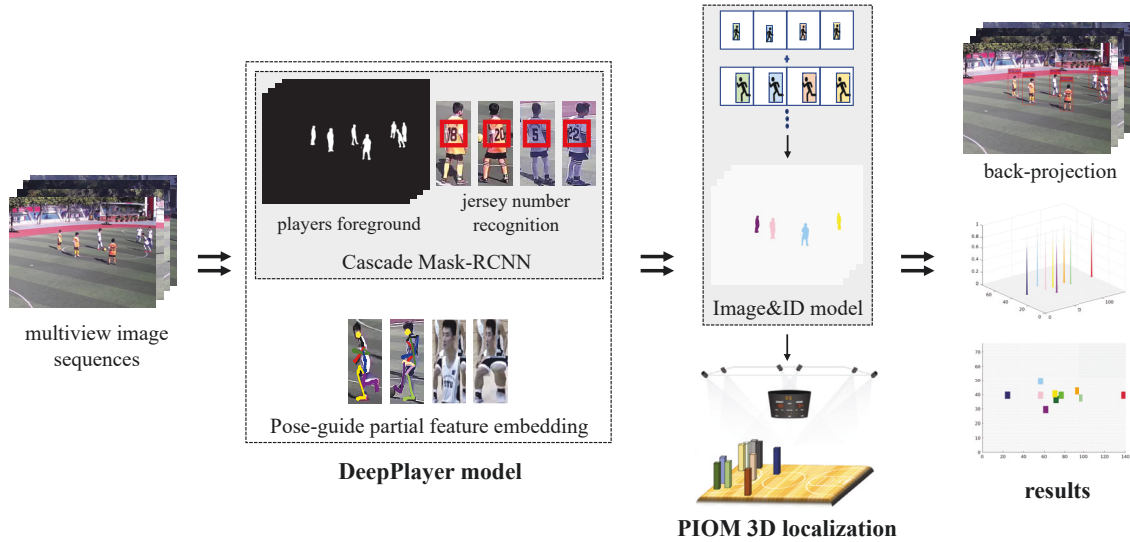


Figure 6.1 : An overview of the PIOM 3D localization framework. Firstly, we use the DeepPlayer model (see Section 5.1) that consists of a Cascade Mask-RCNN model and a PoseID model to extract the sports players’ segmentation masks and identification labels at pixel-level. At the same time, with the 3D world coordinate settings for the sports space mentioned in Section 3.2, we introduce an Image&ID model and an image distance norm to fuse the multiview pixel-wise segmentation and ID labels together with their 3D spatial relations. The synthetic images link the occupancy probabilities with the visible and computable image pixels, while the synthetic ID modules associate the identification inputs from all camera views with accurate spatial coordinates. With our proposed PIOM 3D localization algorithms, we then obtain sports players’ 3D locations and their unique ID labels. The localization results are finally given as the probabilities of locations that are occupied by the specifically labeled players. As shown above, different colors refer to different ID labels.

and the outcomes produced by the DeepPlayer model.

Finally, an efficient iterative process is designed to minimize the loss function and obtain the optimal solutions efficiently.

The PIOM localization method generates accurate locations with correct identities for every object that is visible on the image sequences. As the 3D localization outcomes are unique and distinguishable for each player, this method can effectively overcome heavy occlusion in sports game scenarios. Meanwhile, for some extreme conditions such as super heavy occlusion, partial body occlusion, tiny object segmentation, and high moving speed body gestures, the PIOM approach still keeps excellent performance.

6.1 Multi-Dimensional Bayesian Model

For all the discrete locations \mathbf{G} , we initial a set of discrete random variables $\mathbf{X} = \{X_k | k \in \mathbf{G}\}$, where $X_k \in \{0, 1\}$. We let the Boolean random variable X_k represent the presence or the absence of an individual at location k . $X_k = 1$ represents presence, while $X_k = 0$ represents absence. Thus, the probability of a specific location k that is occupied by an individual can be written as

$$P(X_k = 1) \tag{6.1}$$

To describe the target's ID information, We initial a set of discrete random variables $\mathbf{Y} = \{Y_k | k \in \mathbf{G}\}$. Let Y_k denote the index of the ID calculated at the location k .

Thus, at location k , the probability of presence of an individual with ID index $Y_k = \xi$ can be written as:

$$P(X_k = 1, Y_k = \xi) = P(Y_k = \xi | X_k = 1)P(X_k = 1) \tag{6.2}$$

While absence of the location k is the marginal probability $P(X_k = 0)$.

Let $\mathbf{B} = \{B^1, B^2, \dots, B^C\}$ denote the evidence of presence we obtained from the timing synchronized image sequences $\mathbf{I}_t = \{I_t^1, I_t^2, \dots, I_t^C\}$. This evidence of presence contains targets' segmentation, bounding boxes and ID indices. In this given evidence, we define the presence of an individual with ID index $Y_k = \xi$ at location k as the conditional probability

$$P(X_k = 1, Y_k = \xi | \mathbf{B}) \quad (6.3)$$

We introduce two assumptions of independence about the given evidence and the probabilities for the whole locations. Our first assumption is that an individual in the location does not take into account the presence of the other individuals in his vicinity when moving around, which is true as long as avoidance strategies are ignored. This can be formalized as

$$P((X_1, Y_1), (X_2, Y_2), \dots, (X_G, Y_G) | \mathbf{B}) = \prod_k P((X_k, Y_k) | \mathbf{B}) \quad (6.4)$$

The second assumption is that all statistical dependencies between views are due to the presence of individuals in the sports space. This implies that as soon as the presence of all individuals is known, the views become independent.

$$P(B^1, B^2, \dots, B^C | (\mathbf{X}, \mathbf{Y})) = \prod_c P(B^c | (\mathbf{X}, \mathbf{Y})) \quad (6.5)$$

However, for all locations \mathbf{G} , the conditional probability $P((\mathbf{X}, \mathbf{Y}) | \mathbf{B})$ is intractable. By providing the prior probability $P(\mathbf{X}, \mathbf{Y})$ and the likelihood probability $P(\mathbf{B} | (\mathbf{X}, \mathbf{Y}))$, tracking the posterior probability $P((\mathbf{X}, \mathbf{Y}) | \mathbf{B})$ becomes a Bayesian problem.

$$P((\mathbf{X}, \mathbf{Y}) | \mathbf{B}) = \frac{P(\mathbf{X}, \mathbf{Y})P(\mathbf{B} | (\mathbf{X}, \mathbf{Y}))}{P(\mathbf{B})} \quad (6.6)$$

$$= P(\mathbf{X}, \mathbf{Y})P(\mathbf{B} | (\mathbf{X}, \mathbf{Y})) \quad (6.7)$$

6.2 Image&ID Model

In order to visually describe the occupancy of a target with a particular ID on the location k , we introduce an image model and an ID model by creating synthetic images and synthetic ID modules. The synthetic images associate the Boolean variable X_k s with a range of pixels that are visible and computable. At the same time, the synthetic ID modules link the same range of pixels to the ID value Y_k s, which makes it possible to calculate the occupancy and identification probabilities mathematically. Additionally, we provide an approach to compute the distinction (the distance) between two images, also known as the image norm.

For the rectangle at location k in camera view c , we let a synthetic unit image \mathcal{A}_k^c represent the range of the pixels it contains. Every synthetic unit image is unique as its location and size are defined by the camera calibration and the ground plane discretization. We multiply the synthetic unit image of location k with the Boolean variable X_k , and then sum the results of all the locations \mathbf{G} . The values of X_k s are then visually related with a synthetic image in view c , which is $A^c = \bigoplus_k X_k \mathcal{A}_k^c$. \bigoplus means the sum of multiple images. For all camera views with the same frame index, the synthetic images are a set of variables $\mathbf{A} = \{A^1, A^2, \dots, A^C\}$. Examples of the synthetic unit image and synthetic image are shown in Figure 6.2.

Synthetic ID modules intend to describe the identification inputs visually. For the rectangle at location k in camera view c , we let $r = \{r_1, r_2, \dots, r_{MN}\}$ represent all of the pixels it includes. We denote \mathcal{R}_k as the synthetic unit ID at location k to represent the ID attribute of this range of pixels. For the location k back-projected from a number of camera views where it is visible, we give this equation to estimate the synthetic ID module:

$$R_k = \arg \max_{\xi} \frac{\sum_c (\mathcal{R}_k^c \sum_{i=1}^{MN} r_i | \mathcal{R}_k^c = \xi)}{\sum_c (\mathcal{R}_k^c \sum_{i=1}^{MN} r_i)} \quad (6.8)$$

Where M and N denote the resolution of the image. An example of the synthetic

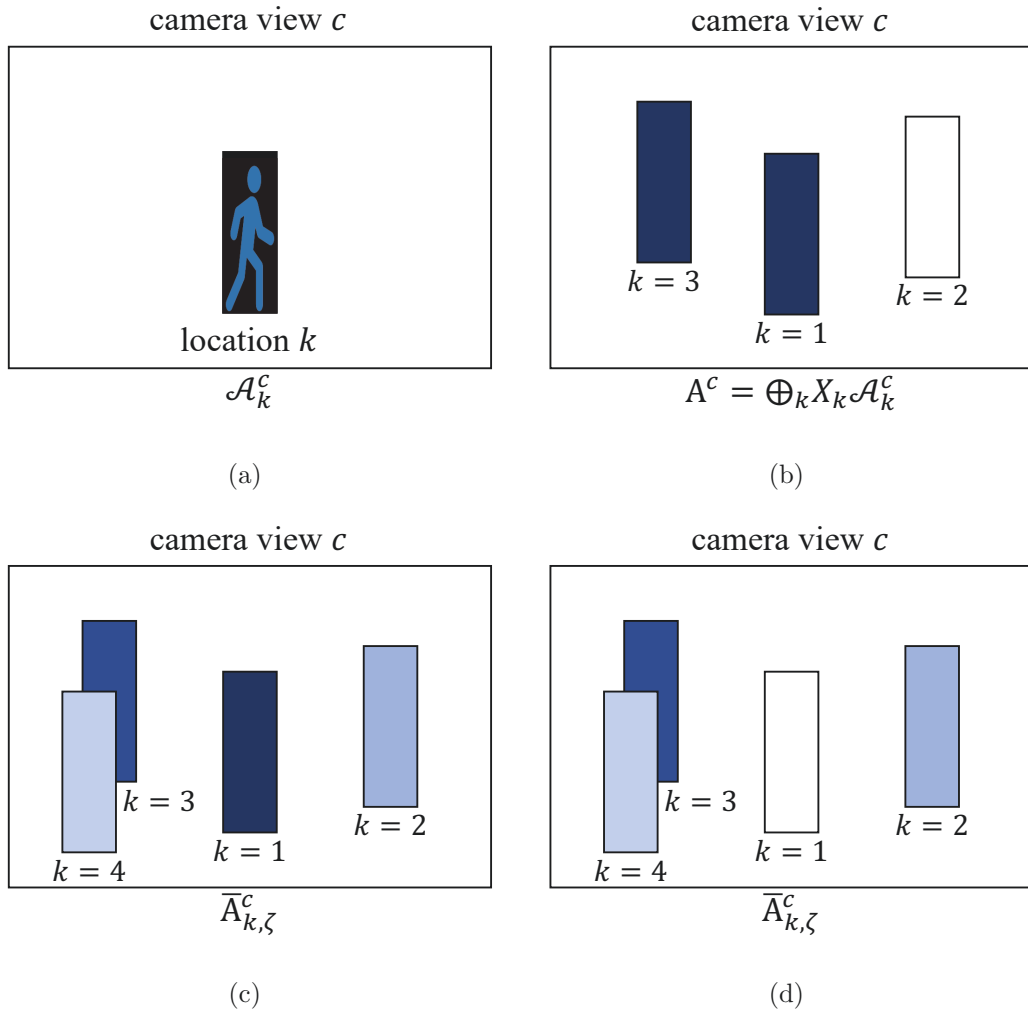


Figure 6.2 : An example of the synthetic unit image, synthetic image and synthetic average images. (a) A synthetic unit image at location k from camera view c , the black area represents the back-projected ground plane grid on that location. (b) indicates the synthetic image where $X_{1,3} = 1$ and $X_2 = 0$. (c) and (d) are 2 examples of the synthetic average images $\bar{A}_{k,\zeta}^c$, when q_k has multiple values: $q_2 = 0.6$, $q_3 = 0.8$, $q_4 = 0.2$. But in (c) $q_1 = 1$, while in (d) $q_1 = 0$.

ID module is shown as Figure 6.3.

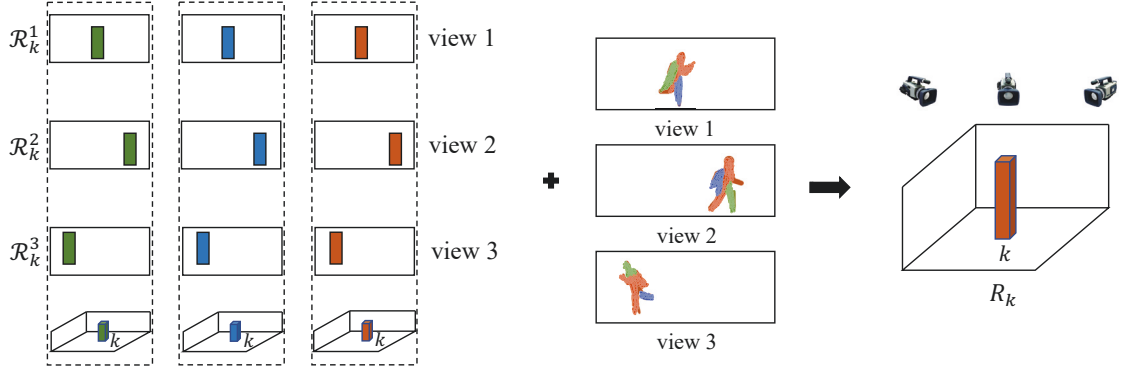


Figure 6.3 : An example of the synthetic ID module. Three different colors refer to 3 different ID attributes. The first column shows the three synthetic unit ID \mathcal{R}_k s with different ID attributes from 3 different views at location k . The second column is the identification inputs from these views that including ID attributes and pixel information. The third column represents the calculated synthetic ID module R_k .

Here we present an efficient approach to compute the similarity and distinction between two images. Let x, y be two M by N images, where $x = (x_1, x_2, \dots, x_{MN})$ and $y = (y_1, y_2, \dots, y_{MN})$. x_{kN+l} and y_{kN+l} represent the gray levels at location (k, l) of the image x, y . We denote $P_i, P_j | i, j = 1, 2, \dots, MN$ the image's pixels. The distance between two pixels is $|P_i, P_j| = \sqrt{(k - k')^2 + (l - l')^2}$, where P_i, P_j are at location $(k, l), (k', l')$. Then we define the distance between two images x, y is

$$D^2(x, y) = \frac{1}{2\pi} \sum_{i,j=1}^{MN} \exp\left(\frac{-|P_i, P_j|^2}{2}\right) (x_i - y_i)(x_j - y_j) \quad (6.9)$$

By using this approach, we can efficiently estimate the errors between the given segmentation and identification inputs with the synthetic images with ID modules, as each pixel that we process includes not only the pixel values but also its ID values.

6.3 The Posterior Probabilities

Let q_k denote the marginal probability at location k . It indicates the probability of presence at location k by a target.

$$q_k = P(X_k = 1, Y_k | \mathbf{B}) \quad (6.10)$$

Let Y_k^* denote the ID value at location k , when $Y_k = \xi$, $P(Y_k)$ achieves the maximum value.

$$Y_k^* = \arg \max_{\xi} P(Y_k = \xi) \quad (6.11)$$

In order to estimate the posterior probability $P(\cdot | \mathbf{B})$, we define a joint distribution Q , that $\mathbf{X}, \mathbf{Y} \sim Q$. We denote E_Q as its expectation. Then we aim to look for a set of marginal probabilities q_k s and the corresponding ID value $Y_k^* | k \in \mathbf{G}$, which can minimize the $K - L$ divergence between the estimation Q we defined and the true posterior probability $P(\cdot | \mathbf{B})$ that we are after. Apply partial derivative of the $K - L$ divergence with respect to the unknown q_k , we have

$$\begin{aligned} \frac{\partial}{\partial q_k} KL(Q || P(\cdot | \mathbf{B})) &= \log \frac{q_k(1 - \varepsilon_k)}{\varepsilon_k(1 - q_k)} + E_Q \left(\sum_c D(B^c, A^c) | X_k = 1, Y_k = R_k \right) \\ &\quad - E_Q \left(\sum_c D(B^c, A^c) | X_k = 0, Y_k = R_k \right) \end{aligned} \quad (6.12)$$

Where D denotes the distance between two images which is defined in Equation (6.9).

Here we provide the differentiation process of partial derivative of the $K - L$ divergence. By applying the partial derivative of the $K - L$ divergence between

these two probability distributions with respect to the target q_k s, we have

$$\begin{aligned} & \frac{\partial}{\partial q_k} KL(Q||P(\cdot|\mathbf{B})) \\ &= \frac{\partial}{\partial q_k} E_Q \left(\log \frac{Q(\mathbf{X}, \mathbf{Y})}{P(\cdot|\mathbf{B})} \right) \end{aligned} \quad (6.13)$$

$$= \frac{\partial}{\partial q_k} E_Q \left(\log \frac{Q(\mathbf{X}, \mathbf{Y})}{P(\mathbf{X}, \mathbf{Y})} + \log P(\mathbf{B}) - \log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) \right) \quad (6.14)$$

$$= \frac{\partial}{\partial q_k} E_Q \left(\sum_l \log \frac{Q(X_l, Y_l)}{P(X_l, Y_l)} - \log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) \right) \quad (6.15)$$

$$= \frac{\partial}{\partial q_k} E_Q \left(\log \frac{Q(X_k, Y_k)}{P(X_k, Y_k)} - \log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) \right) \quad (6.16)$$

$$\begin{aligned} &= \frac{\partial}{\partial q_k} E_Q \left(\log \frac{Q(X_k = 1, Y_k)}{P(X_k = 1, Y_k)} - \log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) | X_k = 1 \right) \\ &+ \frac{\partial}{\partial q_k} E_Q \left(\log \frac{Q(X_k = 0, Y_k)}{P(X_k = 0, Y_k)} - \log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) | X_k = 0 \right) \end{aligned} \quad (6.17)$$

$$\begin{aligned} &\approx \log \frac{q_k}{\varepsilon_k} + 1 - E_Q \left(\log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) | X_k = 1, Y_k = Y_k^* \right) \\ &- \log \frac{1 - q_k}{1 - \varepsilon_k} - 1 + E_Q \left(\log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) | X_k = 0, Y_k = Y_k^* \right) \end{aligned} \quad (6.18)$$

$$\begin{aligned} &= \log \frac{q_k(1 - \varepsilon_k)}{\varepsilon_k(1 - q_k)} - E_Q \left(\log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) | X_k = 1, Y_k = Y_k^* \right) \\ &+ E_Q \left(\log P(\mathbf{B}|\mathbf{X}, \mathbf{Y}) | X_k = 0, Y_k = Y_k^* \right) \end{aligned} \quad (6.19)$$

$$\begin{aligned} &= \log \frac{q_k(1 - \varepsilon_k)}{\varepsilon_k(1 - q_k)} + E_Q \left(\sum_c D(B^c, A^c) | X_k = 1, Y_k = R_k \right) \\ &- E_Q \left(\sum_c D(B^c, A^c) | X_k = 0, Y_k = R_k \right) \end{aligned} \quad (6.20)$$

Equation (6.13) is the definition of the $K - L$ divergence. (6.14) applies the Bayesian model to the term of $P(\cdot|\mathbf{B})$. (6.15) is obtained under the first independence assumption (6.4) we proposed. (6.16) removes the terms that are constant with respect to q_k . We condition the q_k equal to 1 and 0 respectively and obtain (6.17). (6.18) conditions the Y_k equal to Y_k^* and applies the partial derivative, ε_k means the initialized prior probability $P(X_k = 1, Y_k)$ at location k . We replace the term of $\log P(\mathbf{B}|\mathbf{X}, \mathbf{Y})$ with synthetic ID image (6.8) and image distance (6.9) proposed, and apply the second independence assumption (6.5) to obtain (6.20).

Let the partial derivative equals to zero, we can have the optimal set of q_k s that make the estimation Q mostly approximate to the true posterior. We then obtain

$$q_k = \left(1 + \exp \left(\lambda_k + \sum_c E_Q(D(B^c, A^c) | X_k = 1, Y_k = R_k) - E_Q(D(B^c, A^c) | X_k = 0, Y_k = R_k) \right) \right)^{-1} \quad (6.21)$$

Where $\lambda_k = \log \frac{1-\varepsilon_k}{\varepsilon_k}$.

Unfortunately, the calculation of $E_Q(D(B^c, A^c) | X_k = \zeta, Y_k = R_k)$ is intractable. However, since \mathbf{X}, \mathbf{Y} are under the distribution Q and they are independent, we approximate the equation by applying $\zeta \in \{0, 1\}$ and then obtain:

$$E_Q(D(B^c, A^c) | X_k = \zeta, Y_k = R_k) \approx D(B^c, E_Q(A^c | X_k = \zeta, Y_k = R_k)) \quad (6.22)$$

After the approximation we have

$$q_k = \left(1 + \exp \left(\lambda_k + \sum_c D(B^c, E_Q(A^c | X_k = 1, Y_k = R_k)) - D(B^c, E_Q(A^c | X_k = 0, Y_k = R_k)) \right) \right)^{-1} \quad (6.23)$$

$$= \left(1 + \exp \left(\lambda_k + \sum_c D(B^c, \bar{A}_{k,1}^c | Y_k = R_k) - D(B^c, \bar{A}_{k,0}^c | Y_k = R_k) \right) \right)^{-1} \quad (6.24)$$

Where the synthetic average image $\bar{A}_{k,\zeta}^c$ represents the $E_Q(A^c | X_k = \zeta, Y_k = R_k)$, we provide its computation method as

$$\bar{A}_{k,\zeta}^c = E_Q(A^c | X_k = \zeta, Y_k = R_k) \quad (6.25)$$

$$= \sum_{l, l \neq k}^G Q(X_l) \mathcal{A}_l^c | Y_{l, l \neq k} + \zeta \mathcal{A}_k^c | Y_k = R_k \quad (6.26)$$

An example of the synthetic average image $\bar{A}_{k,\zeta}^c$ is shown in Figure 6.2(c) and 6.2(d). As we define the Image&ID model, the likelihood probability $P(\mathbf{B} | (\mathbf{X}, \mathbf{Y}))$

is replaced with a normalized image distance between the given evidence of image sequences and the synthetic average image $\bar{A}_{k,\zeta}^c$ on the condition of the synthetic ID module R_k . The synthetic average image and its ID module are functions of the q_k s and Y_k s. By setting a set of initial values to the q_k s, the synthetic average image with the corresponding ID module is firstly computed. Afterward, we use the outcomes to calculate the distance from the inputted segmentation with identification. Then, the values of q_k s are updated by re-computing the partial derivative of the $K - L$ divergence.

Intuitively, if the synthetic average image computed by q_k s is more similar with the inputted segmentation, and the Y_k^* s fit more with the actual target's identification results, the score $D(B^c, \bar{A}_{k,1}^c | Y_k = R_k)$ decreases while $D(B^c, \bar{A}_{k,0}^c | Y_k = R_k)$ increases, which leading to higher q_k s. With regard to the occlusions, if a rectangle at location k is occluded by another nearby, the computed synthetic unit image at that location would approximate to zero, which makes the q_k remain equal to the prior. While from the other views where the rectangle is clearly visible, the q_k is calculated as usual. Thus, this approach can effectively overcome the heavy occlusions.

6.4 An Efficient Iterative Process

In this subsection, we provide a quick and simple approach to iteratively compute the q_k s and Y_k^* s based on Eq. (6.24) and Eq. (6.8). The iterative procedure is illustrated below.

1. Set a uniform initial value to the q_k s, experimentally we set the initial value to 0.01.
2. For all the locations $k \in \mathbf{G}$, compute its ID value $Y_k^* = R_k$ through every camera, by using Eq. (6.8).

3. This step needs to be done G times as it is for every location each iterative step.
 - (a) For each location k , compute the synthetic average image $\bar{A}_{k,\zeta}^c$ under the condition of $Y_k = R_k$, by using Eq. (6.26).
 - (b) For each location k , compute the distance between the input segmentation and identification $\mathbf{B} = \{B^c | c \in \mathbf{C}\}$ and the synthetic average images $\bar{A}_{k,\zeta}^c$, by using Eq. (6.9).
4. For all the locations k , re-compute the marginal probability q_k s by using Eq. (6.24)
5. Repeat the step 3) and 4) until an optimal solution converges. Usually we do this iteration in an order of 100.

6.5 Experimental Evaluation

In this section, we implement our proposed method PIOM and conduct experiments based on our collected football datasets LH0716v2 and LH0928. We then compare the experimental results with the baseline POM [42] and two of our previous work POM+CNN [34] and PomID [35].

6.5.1 Datasets

We conduct the experiments based on two of our collected football datasets LH0716v2 and LH0928.

The LH0716v2 dataset is a youth football match dataset we collected at Longhu primary school, China. It is a temporal synchronization video and image dataset recorded by eight wide-angle high-resolution cameras. The spatial relation between the football court and the cameras is accurately calibrated. We discretize the football court into 140×76 grids on the ground floor with a size of $500 \times 500mm$. Video files

are acquired at 25 fps and 2560×1920 resolution in the format of MPEG-4. There are ten players, including two keepers, that are labeled using unique identifications. We implement our localization framework over 10,000 frames and collect the results comparing with the previous methods mentioned above.

The LH0928 dataset is collected at the same football field as LH0716v2 but with different parameter settings and different football games. It is recorded by eight 1920×1080 cameras with 16 players labeled by unique identities. We also conduct experiments on around 10,000 image frames.

6.5.2 Metrics and baselines

We give the experimental results as the probabilities of locations occupied by players with specific identifications, which are very peaky. We therefore simply treat the location where the probability of presence is higher than 0.75 as a proposal. Comparing selected proposals with the given ground-truth locations, we use bird-eye view distance (BV), 2D IoU (IoU_{2d}), and 3D IoU (IoU_{3d}) as three thresholds to select positive results.

Practically, as these two football datasets discretize the football court with many $500 \times 500mm$ grids on the ground plane, the distance between two neighboring locations is set to be $500mm$, which is a usual distance when two football players are standing close. Consequently, we usually set the three thresholds to be $500mm$, 0.5, and 0.25 to meet the criterion when selecting proposals. By applying the thresholds with specific values, we count all produced proposals and obtain the number of correct proposals as true-positive (TP), missing proposals as false-negative (FN), and incorrect proposals as false-positive (FP). Note that we can apply various values to the three thresholds to obtain different sets of TP, FN, FP proposals. Given these, we can evaluate Precision/Recall, MODA, and MODP mentioned in Section 4.3. Additionally, we can also evaluate the Average Precision (AP) proposed by

PASCAL VOC [94].

We will report P/R, MODA, MODP, and AP results under multiple threshold settings. Note that these metrics are unforgiving of projection errors. Nevertheless, we believe them to be the metrics for a multi-camera system that computes the 3D location for objects.

In order to compare our proposed method PIOM, we implement the following three baselines:

- **POM** [42], a traditional multi-camera multi-pedestrian detector that simply uses the Gaussian mixture model to generate overall foreground masks for multiple pedestrians.
- **POM+CNN** [34], a multi-camera multi-pedestrian detector developed from the traditional POM method (see Chapter 4). Instead of the Gaussian model, CNN was used to generate foreground subtraction for all pedestrians as the input of the algorithms. For those two football datasets mentioned above, we do not implement the IniSet scheme at this stage because fisheye cameras are unavailable.
- **PomID** [35], our proposed multi-camera multi-player localization method in Chapter 5, extending 3D detection to more challenging sport players scenarios. This approach uses coarse-to-fine number recognition and pose-guided partial feature embedding to generate both foreground masks and ID labels for each player individually.

As two of the baselines POM and POM+CNN only process pixel-wise foreground masks and do not take into account the players' ID labels, we use different protocols to calculate the TP, FN, and FP proposals for the baseline PomID and our proposed approach PIOM.

For POM and POM+CNN:

- TP, the calculated bird-eye distance is lower than, or 2/3D IoU is higher than the given threshold;
- FN, no generated proposal nearby the given ground-truth location;
- FP, the calculated bird-eye distance is higher than, or 2/3D IoU is lower than the given threshold;

While for PomID and PIOM:

- TP, must meet both of the following two conditions:
 1. the ID label is identical;
 2. the calculated bird-eye distance is lower than, or 2/3D IoU is higher than the given threshold;
- FN, must meet one of the following two conditions:
 1. no generated proposal nearby the given ground-truth location;
 2. the generated proposal has unconfirmed ID label;
- FP, must meet one of the following two conditions:
 1. the ID label is not identical;
 2. the calculated bird-eye distance is higher than, or 2/3D IoU is lower than the given threshold;

6.5.3 Algorithm implementation and configuration

We train the DeepPlayer model with SGD solver [95] in three steps, and follow the image-centric sampling strategy in [17]. As shown in Table 6.1, the training

sequence starts with Cascade Mask-RCNN-P, followed by Cascade Mask-RCNN-J, after which is PoseID. We use the ResNet-50 pre-trained by the ImageNet 1000-class dataset. Other layers are randomly initialized by a Gaussian distribution with a standard deviation of 0.01 and a mean of 0. The ratio of positive and negative anchors in each image is set as 1 : 3. In Eq. (5.3), we set $\lambda_1 = 0.55$, $\lambda_2 = 0.45$, experimentally. On each GPU, the mini-batch size is 2. The whole training takes 20 hours on four NVIDIA 1080Ti Pascals under the Caffe framework.

	Step size	Learning rate	γ	Momentum	Weight decay
Cascade Mask-RCNN-P	80,000	0.002	0.1	0.9	0.001
Cascade Mask-RCNN-J	60,000	0.001	0.1	0.9	0.0005
PoseID	40,000	0.05	0.1	0.9	0.0005

Table 6.1 : Training parameters of the DeepPlayer model in three steps.

In order to mathematically compute the Image&ID model, we develop an IdMap scheme to extend the dimension of every input pixel. This scheme makes each pixel contain not only foreground/background information but also distinguished ID values. This model processes and stores both binary pixel values and multiple ID values with the fixed dimension of width and height, which equivalent to the original image sequences. For the two datasets mentioned above, experimentally, we set the typical human height to $160mm$, use 10 and 16 scalar values to retrieve ID labels for the two datasets, respectively.

6.5.4 Results

We implement our method on the LH0716v2 dataset and back-project the generated locations onto original image sequences, shown in Figure 6.4. The four columns show examples of four frames 221, 241, 261, and 281 of camera 1, 4, 5, and 8.



Figure 6.4 : Illustration of our localization results implemented on LH0716v2 dataset. The red cube refers to the generated location, while the yellow label indicates the detected identification outcome. In row No. 5 and 6, different colors refer to unique ID labels. Row No. 5 illustrates the localization results in 3D space, while No. 6 shows the results on the ground plane of bird-eye view.

As can be seen from the first four rows, red cubes refer to the generated locations, and yellow labels indicate the detected identification for each player. The fifth row shows the 3D localization results with various colors, which are plotted according to the calculated probabilities. Each color indicates one unique identification output. The last row shows the heat-map of the 3D localization on the ground plane of bird-eye view. Dots with different colors refer to the detected locations with specific ID labels.

From this example, we can see that our method generates accurate locations with correct identification outputs for every player that is visible on the image sequences. As our 3D localization outcomes are unique and distinguishable for each player, our method can effectively overcome heavy occlusion in sport game scenarios. Additionally, our approach can also avoid ID switches (see the yellow and green locations in the last row). Thus our method can help improve the performance of some multi-object trackers which rely on tracking-by-detecting strategies.

In Figure 6.5, we implement our method based on the LH0928 dataset. Three rows in the figure illustrate the back-projection of generated results from camera 2, 3, 6, and 7. It shows that our proposed method obtains accurate localization results and correct ID labels for each player on this dataset.

We then selected several typical instances to illustrate that our proposed localization method can overcome some extreme negative conditions, such as extreme heavy occlusion, partial body occlusion, tiny player segmentation from video cameras, and high moving speed body gestures, as these conditions are considerably common in multiple sports players detection and localization scenarios. These examples are implemented on the LH0716v2 dataset, as shown in Figure 6.6.

The first and third columns are the original back-projection of localization results, and we zoom the crowded areas of the images (within green borders) to column

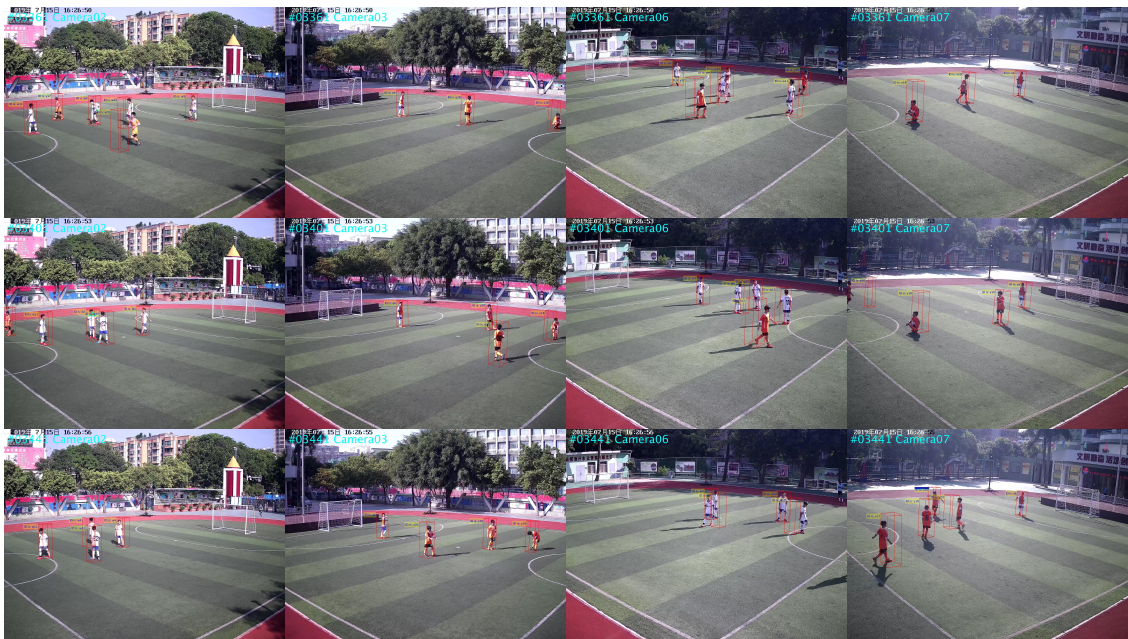


Figure 6.5 : Illustration of our localization results implemented on LH0928 dataset. Three rows indicate back-projection of the localization results from frame 3361, 3401, and 3441 on camera 2, 3, 6, and 7. The red cube refers to the generated location, while the yellow label indicates the detected identification outcome.

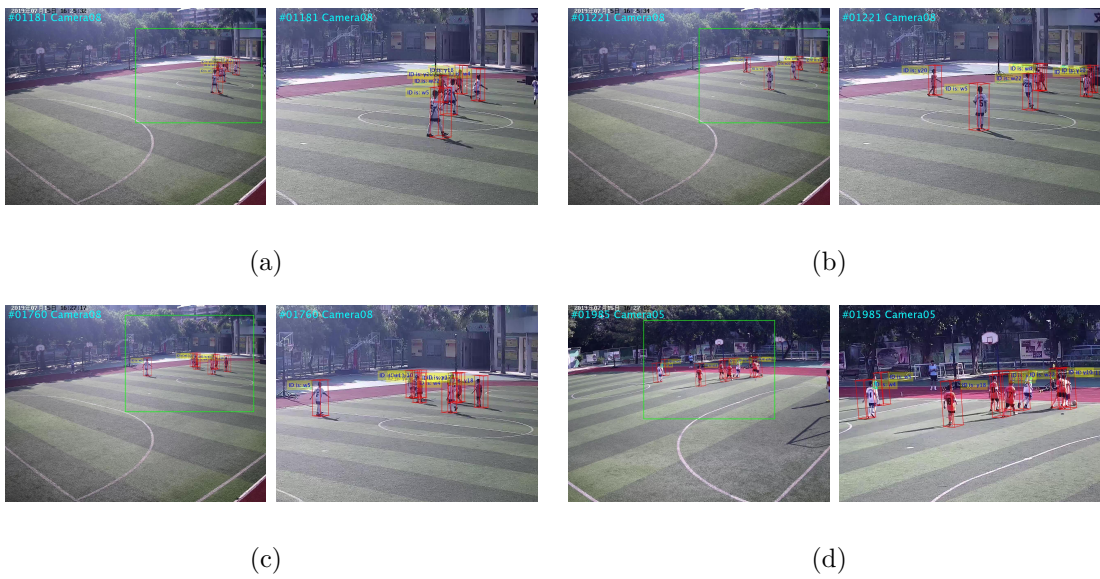


Figure 6.6 : Illustration of our localization results implemented on LH0716v2 dataset that can overcome some extreme negative conditions, such as extremely crowded scenes, full/partial body occlusion, and inaccurate 2D detection from tiny objects. The first and third columns show the original back-projection of our localization results, while the second and fourth columns are obtained by zooming the crowded scenes inside the green borders.

No. 2 and 4, respectively. From the second and fourth columns, we can see that in these extremely crowded scenes, the football players are typically fully or partially occluded by each other. It is significantly difficult to obtain complete foreground masks and correct identification, thus causes negative effects for 2D detection and 3D localization.

Additionally, in the original views, the players are standing far away from cameras, which causes tiny and incomplete 2D detection results. This also brings negative impacts. While, as shown from the back-projection, our proposed approach can effectively obtain correct and accurate locations and identification, eliminating the impacts from full/partial body occlusion, tiny 2D detection results, and high-speed moving players.

We afterward report the curves of Precision/Recall under various threshold settings that mentioned above, as shown in Figure 6.7. Our method is implemented on the LH0716v2 dataset and is compared with the PomID approach. Specifically, we set the thresholds BV , IoU_{2d} and IoU_{3d} to the range of $200 \sim 2000mm$, $0.05 \sim 0.95$, $0.05 \sim 0.95$ respectively.

As can be seen in Figure 6.7 (a) and (d), our proposed method has a higher index under all BV distance selections. Especially when distance equals to $500mm$, which roughly corresponds to the width of a human body, our proposed method outperforms the PomID approach by a large margin. In Figure 6.7 (b) and (e), we can see that when IoU_{2d} is lower than 0.65, our proposed method achieves higher scores while precision and recall drop dramatically when the threshold is higher than 0.65. Although PomID performs better when IoU_{2d} is higher than 0.65, precision and recall drop lower than 0.3, which is not practically acceptable.

From Figure 6.7 (c) and (f) we can obtain that, when IoU_{3d} equals 0.25, which can represent average sports players distance, our proposed method outperforms

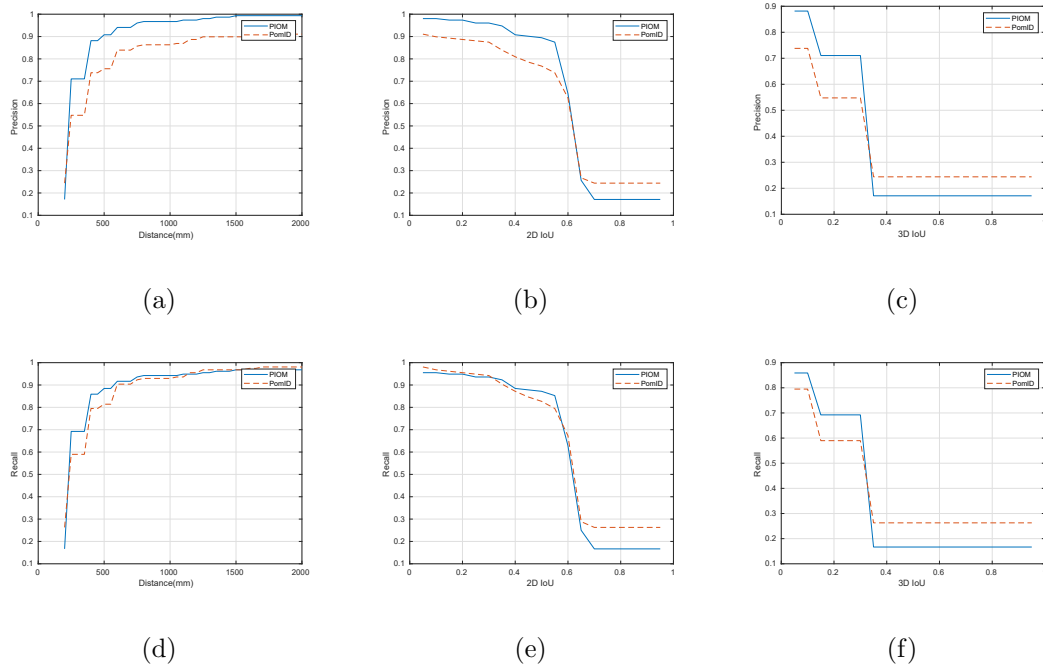


Figure 6.7 : Precision/Recall curves under multiple threshold settings. Results are obtained by implementing both PomID and our PIOM method on the LH0716v2 dataset. (a) and (d), P/R curves under BV distances range from $200 \sim 2000mm$; (b) and (e), P/R curves under IoU_{2d} ratios range from $0.05 \sim 0.95$; (c) and (f), P/R curves under IoU_{3d} ratios range from $0.05 \sim 0.95$. It illustrates that our proposed method PIOM outperforms the previous method PomID generally.

primarily than PomID. PomID also performs better when the scores drop under 0.25, but this score is not applicable when selecting positive proposals. Note that the curves are discrete because we use the 3D cubes to calculate IoU_{3d} ratios.

Implementing the three baselines POM, POM+CNN, PomID and our proposed method PIOM on the LH0716v2 dataset, we then report Precision/Recall and MODA/MODP scores under various thresholds selections, which are: (1) BV equals to 250mm and 500mm; (2) IoU_{2d} ratio equals to 0.5 and 0.75; (3) IoU_{3d} ratio equals to 0.25 and 0.5, respectively. The scores are shown in Table 6.2 and Table 6.3.

Method	Precision/Recall(%)					
	bv=250mm	bv=500mm	$IoU_{2d} = 0.5$	$IoU_{2d} = 0.75$	$IoU_{3d} = 0.25$	$IoU_{3d} = 0.5$
POM	33.60/39.45	53.60/58.45	58.27/68.85	18.27/20.85	40.69/47.83	18.69/20.83
POM+CNN	42.48/46.50	69.48/70.50	70.65/75.82	20.65/22.82	51.87/53.66	21.87/22.66
PomID	48.75/52.41	75.60/81.41	76.79/82.69	22.79/28.69	54.76/58.97	24.76/26.97
PIOM	51.85/58.46	90.79/88.46	89.47/87.18	19.47/19.18	71.05/69.23	18.05/18.23

Table 6.2 : Quantitative comparison results of the proposed method with the other three baselines on Precision/Recall scores, implemented on the LH0716v2 dataset.

Method	MODA/MODP(%)					
	bv=250mm	bv=500mm	$IoU_{2d} = 0.5$	$IoU_{2d} = 0.75$	$IoU_{3d} = 0.25$	$IoU_{3d} = 0.5$
POM	25.76/47.50	45.76/37.60	40.55/39.32	20.55/45.72	26.22/45.08	20.22/56.84
POM+CNN	31.75/49.24	47.75/39.75	46.35/40.60	26.65/53.40	30.75/48.75	26.75/59.43
PomID	34.06/57.25	52.22/41.88	48.91/44.53	32.46/72.67	34.06/57.25	32.46/72.67
PIOM	37.16/53.45	52.57/41.13	49.89/43.07	31.05/70.67	37.16/53.45	31.05/70.67

Table 6.3 : Quantitative comparison results of the proposed method with the other three baselines on MODA/MODP scores, implemented on the LH0716v2 dataset.

From Table 6.2, our proposed method outperforms all the other approaches with respect to the two BV selections, $IoU_{2d} = 0.5$ and $IoU_{3d} = 0.25$. It proves that our

proposed method achieves state-of-the-art performance in general scenarios. Specifically, when we choose the BV , IoU_{2d} and IoU_{3d} to be $500mm$, 0.5 and 0.25 , which are practically applicable for 3D localization algorithm evaluation with discretized location definition, our proposed method achieves the best performance than any other methods by a large margin.

In Table 6.3, our proposed method generally achieves the best performance with respect to MODA and MODP scores. It demonstrates that our proposed method has more accurate system performance and a higher overlapping ratio with respect to the positive locations. In specific, the MODA scores of our proposed method reach 0.52 , 0.49 , and 0.37 with the threshold settings mentioned above, which is considerably higher, although we use strict constraints to report TP, FN and FP. Evaluation of these four metrics proves that our proposed method has a more accurate and robust performance than the other three baselines.

We also report various Average Precision (AP) scores under nine sets of threshold settings by implementing the baselines and our proposed method on both LH0716v2 and LH0928 datasets, as shown in Table 6.4 and Table 6.5. The threshold settings are: (1) $BV = 250, 500, 750mm$; (2) $IoU_{2d} = 0.25, 0.5, 0.75$; (3) $IoU_{3d} = 0.25, 0.5, 0.75$. Here we treat AP_{bv}^{500} , $AP_{2d}^{0.5}$ and $AP_{3d}^{0.25}$ as applicable mode, AP_{bv}^{250} , $AP_{2d}^{0.75}$ and $AP_{3d}^{0.5}$ as hard mode, because for 3D localization algorithm implementation in discretized sport space, the applicable mode is qualified enough to evaluate the system performance, while the results of hard mode are considerably limited by strict constrains.

As reported from Table 6.4 and Table 6.5, our proposed method outperforms the other three baselines by a large margin, especially for the applicable mode, reaching the number of $0.48/0.47$, $0.78/0.76$, and $0.61/0.61$ for both datasets. Even for the hard mode, our proposed method is proved to have better AP scores. It

demonstrates that our proposed method has better overall 3D localization accuracy under both 2D and 3D metrics.

LH0716v2 dataset									
Method	AP(bird-view)(%)			AP(IoU_{2d})(%)			AP(IoU_{3d})(%)		
	AP_{bv}^{250}	AP_{bv}^{500}	AP_{bv}^{750}	$AP_{2d}^{0.25}$	$AP_{2d}^{0.5}$	$AP_{2d}^{0.75}$	$AP_{3d}^{0.25}$	$AP_{3d}^{0.5}$	$AP_{3d}^{0.75}$
POM	20.86	32.47	39.88	78.66	59.45	16.83	46.18	16.44	12.00
POM+CNN	24.89	37.93	49.05	85.55	68.24	18.95	51.14	19.15	18.23
PomID	28.03	39.09	49.73	92.61	71.10	19.34	51.56	19.34	18.56
PIOM	35.00	47.92	57.91	92.36	78.18	31.17	61.30	26.20	20.15

Table 6.4 : Quantitative comparison results of the proposed method with the other baselines on AP scores, implemented on the LH0716v2 dataset.

LH0928 dataset									
Method	AP(bird-view)(%)			AP(IoU_{2d})(%)			AP(IoU_{3d})(%)		
	AP_{bv}^{250}	AP_{bv}^{500}	AP_{bv}^{750}	$AP_{2d}^{0.25}$	$AP_{2d}^{0.5}$	$AP_{2d}^{0.75}$	$AP_{3d}^{0.25}$	$AP_{3d}^{0.5}$	$AP_{3d}^{0.75}$
POM	20.16	31.45	40.22	79.01	58.66	15.33	44.63	14.27	10.20
POM+CNN	24.44	37.91	49.65	85.66	67.82	18.64	51.40	18.91	17.22
PomID	26.64	37.96	49.02	92.24	70.18	17.98	52.03	18.18	15.66
PIOM	33.98	47.15	58.55	92.88	76.84	28.31	61.08	24.67	18.75

Table 6.5 : Quantitative comparison results of the proposed method with the other baselines on AP scores, implemented on the LH0928 dataset.

6.6 Conclusion

In this chapter, we proposed the PIOM 3D localization method. This approach uses the DeepPlayer model that consists of a Cascade Mask-RCNN model and a PoseID model to extract the sports players' segmentation and identification labels

at the pixel level. At the same time, we introduce an Image&ID model and an image distance norm to fuse the multiview pixel-wise segmentation and ID labels together with their 3D spatial relations. The Image&ID model consists of a set of synthetic images and synthetic ID modules. The synthetic images link the occupancy probabilities with the visible and computable image pixels, while the synthetic ID modules associate the identification inputs from all camera views with accurate spatial coordinates. Afterward, we develop a multi-dimensional Bayesian model and then construct a loss function as the $K - L$ divergence between an estimated probability distribution and the true posterior probability. The prior probabilities are initialized at the beginning of the iteration, while the likelihood probabilities are approximated by the normalized image distances between the synthetic average images with ID modules and the outcomes produced by the DeepPlayer model. Finally, an efficient iterative process is designed to minimize the loss function and obtain the optimal solutions efficiently.

We then implement our proposed method PIOM and conduct experiments based on our collected football datasets LH0716v2 and LH0928. We then compare the experimental results with the baseline POM and two of our previous methods POM+CNN and PomID. Experimental evaluation demonstrates that our proposed method outperforms these three baselines.

The PIOM localization method generates accurate locations with correct identities for every object that is visible on the image sequences. As the 3D localization outcomes are unique and distinguishable for each player, this method can effectively overcome heavy occlusion in sports game scenarios. Meanwhile, for some extreme conditions such as super heavy occlusion, partial body occlusion, tiny object segmentation, and high moving speed body gestures, the PIOM approach still keeps excellent performance.

Chapter 7

Conclusion and Discussion

7.1 Conclusion

In this thesis, we conduct comprehensive research to solve the problems of multiple-camera multiple-object 3D localization for sports video scenarios.

Firstly, we proposed the POM+CNN+IniSet 3D localization method. This approach applies the CNN-based monocular object detection method jointly on multiple cameras to generate clear and correct foreground masks. We use those foreground masks containing players' segmentation and coordinates to replace pixel-wise binary background subtraction. As a result, ambiguous and false-positive detection results caused by unclear foreground input are successfully eliminated. Moreover, we take advantage of two fisheye cameras arranged above the head of the sports court and develop a generic Bayesian model to initialize a set of indicative parameters. We use the foreground masks produced by the two fisheye cameras to pre-define those parameters with higher or lower initial values. This method enriches the localization system's input information, removes undesired segmentation masks, improves the precision and recall of the localization outcomes, and boosts the localization performance.

Afterward, we proposed the PomID 3D localization method. This approach applies a DeepPlayer model including a Cascade Mask-RCNN model and a pose-guided partial feature embedding to conduct object segmentation and identification for multiple sports objects. The DeepPlayer model produces both the individuals' foreground masks and their identities, which are treated as the given evidence for

the 3D localization algorithms. This method then separately estimates the likely location for each player who has certain and correct identity input and jointly calculates the results for the rest targets without ID labels. Final outcomes are then refined by a set of reasonable constraints. The PomID method includes multiple objects’ identities as evidence to estimate the likely occupied locations, making the localization results distinguishable and unique to be associated with the particular objects. This method can accurately locate multiple objects and effectively avoid identity switches for multiple-object detection and tracking tasks.

Finally, we proposed the PIOM 3D localization method. This approach uses the DeepPlayer model that consists of a Cascade Mask-RCNN model and a PoseID model to extract the sports players’ segmentation and identification labels at the pixel level. At the same time, we introduce an Image&ID model and an image distance norm to fuse the multiview pixel-wise segmentation and ID labels together with their 3D spatial relations. The Image&ID model consists of a set of synthetic images and synthetic ID modules. The synthetic images link the occupancy probabilities with the visible and computable image pixels. The synthetic ID modules associate the identification inputs from all camera views with accurate spatial coordinates. Afterward, we develop a multi-dimensional Bayesian model and construct a loss function as the $K - L$ divergence between an estimated probability distribution and the true posterior probability. The prior probabilities are initialized at the beginning of the iteration, while the likelihood probabilities are approximated by the normalized image distances between the synthetic average images with ID modules and the outcomes produced by the DeepPlayer model. Finally, an efficient iterative process is designed to minimize the loss function and obtain the optimal solutions efficiently. The PIOM localization method generates accurate locations with correct identities for every object that is visible on the image sequences. As the 3D localization outcomes are unique and distinguishable for each player, this method

can effectively overcome heavy occlusion in sports game scenarios. Meanwhile, for some extreme conditions such as super heavy occlusion, partial body occlusion, tiny object segmentation, and high moving speed body gestures, the PIOM approach still keeps excellent performance.

7.2 Discussion

There is still some limitation of our proposed 3D localization frameworks. The POM+CNN+IniSet method does not apply to the situation without fisheye camera installation. Fisheye cameras are more common in basketball datasets than other sports games because the installation is more effortless in basketball gyms. Thus this method cannot be applied in all sports scenarios. The PomID method requires high CPU and GPU capacity and is extremely time-consuming to process sports image sequences. Because sports video datasets usually require high fps configuration. At the same time, the performance of the PIOM localization framework is sensitive to the quality of estimated ID proposals. This method does not take into account the depth information from RGB-D sensors, which are widely used recently.

The remaining challenges of MCMOL tasks still need to be solved. For the basketball datasets, occlusion tends to be extremely severe because it is very common that several players are standing close in tiny areas. Segmentation masks produced by CNN do not usually contain the lower half of the players' bodies occluded by others. Those masks would cause incorrect localization outcomes. Meanwhile, for some sports games that require large sports courts, the cameras are usually installed far away from the players. This would produce tiny bodies captured by the cameras, making it considerably challenging to recognize the players' identities. Occluded segmentation masks are also difficult to be processed.

In our future work, we will extend our proposed 3D localization frameworks to address the current issues discussed above. We will apply our frameworks to more

sports video datasets such as football and volleyball. The experimental configuration, such as multi-camera installation, usually varies from different sports games. Meanwhile, we will take into account the depth information from RGB-D image sequences to extend our 3D localization framework. The depth information will be processed at the same time as the segmentation and identification procedures. The localization algorithms will be re-designed to produce the exact 3D coordinates other than location encoding. We will also consider using point clouds datasets in the future. Point cloud-based methods for autonomous driving have achieved satisfying performance recently. We will construct point cloud-based sports video datasets and conduct research in point clouds methods for sports video scenarios.

Bibliography

- [1] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, “Audio keywords generation for sports video analysis,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 4, no. 2, pp. 1–23, 2008.
- [2] H.-C. Shih, “A survey of content-aware video analysis for sports,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 5, pp. 1212–1231, 2017.
- [3] C.-M. Chen and L.-H. Chen, “Novel framework for sports video analysis: A basketball case study,” in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 961–965.
- [4] T. D’Orazio and M. Leo, “A review of vision-based systems for soccer video analysis,” *Pattern recognition*, vol. 43, no. 8, pp. 2911–2926, 2010.
- [5] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli, “See more, know more: Unsupervised video object segmentation with co-attention siamese networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 3623–3632.
- [6] W. Wang, J. Shen, F. Porikli, and R. Yang, “Semi-supervised video object segmentation with super-trajectories,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 4, pp. 985–998, 2018.
- [7] J. Boisvert, M.-A. Drouin, and P.-M. Jodoin, “High-speed transition patterns for video projection, 3d reconstruction, and copyright protection,” *Pattern Recognition*, vol. 48, no. 3, pp. 720–731, 2015.

- [8] T. Pribanić, T. Petković, and M. onlić, “3d registration based on the direction sensor measurements,” *Pattern Recognition*, vol. 88, pp. 532–546, 2019.
- [9] Y. Lu and S. An, “Research on sports video detection technology motion 3d reconstruction based on hidden markov model,” *Cluster Computing*, vol. 23, no. 3, pp. 1899–1909, 2020.
- [10] W. Wang, J. Shen, X. Dong, A. Borji, and R. Yang, “Inferring salient objects from human fixations,” *IEEE transactions on pattern analysis and machine intelligence*, 2019.
- [11] W. Wang, J. Shen, and F. Porikli, “Saliency-aware geodesic video object segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3395–3402.
- [12] W. Wang, J. Shen, J. Xie, and F. Porikli, “Super-trajectory for video segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1671–1679.
- [13] C. Szegedy, A. Toshev, and D. Erhan, “Deep neural networks for object detection,” 2013.
- [14] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.
- [16] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

- [17] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *arXiv preprint arXiv:1506.01497*, 2015.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*. Springer, 2016, pp. 21–37.
- [20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.
- [21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [22] J. Redmon and A. Farhadi, “Yolo9000: better, faster, stronger,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.
- [23] Z. Zou, Z. Shi, Y. Guo, and J. Ye, “Object detection in 20 years: A survey,” *arXiv preprint arXiv:1905.05055*, 2019.
- [24] P. Jiménez, F. Thomas, and C. Torras, “3d collision detection: a survey,” *Computers & Graphics*, vol. 25, no. 2, pp. 269–285, 2001.
- [25] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, “A survey on 3d object detection methods for autonomous driving applications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3782–3795, 2019.

- [26] Z. Qin, J. Wang, and Y. Lu, “Monogrnet: A geometric reasoning network for monocular 3d object localization,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8851–8858.
- [27] S. Song and M. Chandraker, “Joint sfm and detection cues for monocular 3d localization in road scenes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3734–3742.
- [28] M. Liang, B. Yang, S. Wang, and R. Urtasun, “Deep continuous fusion for multi-sensor 3d object detection,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 641–656.
- [29] Y. Xu, X. Liu, Y. Liu, and S.-C. Zhu, “Multi-view people tracking via hierarchical trajectory composition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4256–4265.
- [30] P. Baqué, F. Fleuret, and P. Fua, “Deep occlusion reasoning for multi-camera multi-target detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 271–279.
- [31] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-view 3d object detection network for autonomous driving,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1907–1915.
- [32] P. Li, X. Chen, and S. Shen, “Stereo r-cnn based 3d object detection for autonomous driving,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7644–7652.
- [33] A. Simonelli, S. R. Buló, L. Porzi, M. López-Antequera, and P. Kotschieder, “Disentangling monocular 3d object detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1991–1999.

- [34] Y. Yang, M. Xu, W. Wu, R. Zhang, and Y. Peng, “3d multiview basketball players detection and localization based on probabilistic occupancy,” in *2018 Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2018, pp. 1–8.
- [35] R. Zhang, L. Wu, Y. Yang, W. Wu, Y. Chen, and M. Xu, “Multi-camera multi-player tracking with deep player identification in sports video,” *Pattern Recognition*, vol. 102, p. 107260, 2020.
- [36] Q. Liang, W. Wu, Y. Yang, R. Zhang, Y. Peng, and M. Xu, “Multi-player tracking for multi-view sports videos with improved k-shortest path algorithm,” *Applied Sciences*, vol. 10, no. 3, p. 864, 2020.
- [37] A. Torralba, K. P. Murphy, and W. T. Freeman, “Sharing visual features for multiclass and multiview object detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 854–869, 2007.
- [38] J. Liebelt and C. Schmid, “Multi-view object class detection with a 3d geometric model,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1688–1695.
- [39] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
- [40] Q. Zhang and R. Pless, “Extrinsic calibration of a camera and laser range finder (improves camera calibration),” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3. IEEE, 2004, pp. 2301–2306.
- [41] R. Eshel and Y. Moses, “Homography based multiple camera detection and

- tracking of people in a dense crowd,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.
- [42] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, “Multicamera people tracking with a probabilistic occupancy map,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 2, pp. 267–282, 2007.
- [43] S. M. Khan and M. Shah, “Tracking multiple occluding people by localizing on multiple scene planes,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 3, pp. 505–519, 2008.
- [44] P. Peng, Y. Tian, Y. Wang, J. Li, and T. Huang, “Robust multiple cameras pedestrian detection with multi-view bayesian network,” *Pattern Recognition*, vol. 48, no. 5, pp. 1760–1772, 2015.
- [45] C. Rubino, M. Crocco, and A. Del Bue, “3d object localisation from multi-view image detections,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1281–1294, 2017.
- [46] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka, “3d bounding box estimation using deep learning and geometry,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7074–7082.
- [47] H. Wang, Y. Yan, J. Hua, Y. Yang, X. Wang, X. Li, J. R. Deller, G. Zhang, and H. Bao, “Pedestrian recognition in multi-camera networks using multilevel important salient feature and multcategory incremental learning,” *Pattern Recognition*, vol. 67, pp. 340–352, 2017.
- [48] T. Chavdarova, P. Baqué, S. Bouquet, A. Maksai, C. Jose, T. Bagautdinov, L. Lettry, P. Fua, L. Van Gool, and F. Fleuret, “Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5030–5039.

- [49] S. Song and J. Xiao, “Sliding shapes for 3d object detection in depth images,” in *European conference on computer vision*. Springer, 2014, pp. 634–651.
- [50] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, “Frustum pointnets for 3d object detection from rgb-d data,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 918–927.
- [51] Y. You, Y. Wang, W.-L. Chao, D. Garg, G. Pleiss, B. Hariharan, M. Campbell, and K. Q. Weinberger, “Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving,” *arXiv preprint arXiv:1906.06310*, 2019.
- [52] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, “Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8445–8453.
- [53] M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner, “Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1355–1361.
- [54] Y. Zhou and O. Tuzel, “Voxelnet: End-to-end learning for point cloud based 3d object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4490–4499.
- [55] S. Shi, X. Wang, and H. Li, “Pointcnn: 3d object proposal generation and detection from point cloud,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 770–779.
- [56] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, “From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network,” *IEEE transactions on pattern analysis and machine intelligence*, 2020.

- [57] W. Shi and R. Rajkumar, “Point-gnn: Graph neural network for 3d object detection in a point cloud,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1711–1719.
- [58] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel, “On the segmentation of 3d lidar point clouds,” in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 2798–2805.
- [59] L. Stelmach, W. J. Tam, D. Meegan, and A. Vincent, “Stereo image quality: effects of mixed spatio-temporal resolution,” *IEEE Transactions on circuits and systems for video technology*, vol. 10, no. 2, pp. 188–193, 2000.
- [60] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, and M.-H. Yang, “Deep regression tracking with shrinkage loss,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 353–369.
- [61] R. Zhang, C. Mu, Y. Yang, and L. Xu, “Research on simulated infrared image utility evaluation using deep representation,” *Journal of Electronic Imaging*, vol. 27, no. 1, p. 013012, 2018.
- [62] R. Zhang, C. Mu, M. Xu, L. Xu, Q. Shi, and J. Wang, “Synthetic ir image refinement using adversarial learning with bidirectional mappings,” *IEEE Access*, vol. 7, pp. 153 734–153 750, 2019.
- [63] W. Ge and R. T. Collins, “Crowd detection with a multiview sampler,” in *European Conference on Computer Vision*. Springer, 2010, pp. 324–337.
- [64] A. Utasi and C. Benedek, “A 3-d marked point process model for multi-view people detection,” in *CVPR 2011*. IEEE, 2011, pp. 3385–3392.

- [65] K.-H. Lo, C.-J. Wang, J.-H. Chuang, and H.-T. Chen, “Acceleration of vanishing point-based line sampling scheme for people localization and height estimation via 3d line sampling,” in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. IEEE, 2012, pp. 2788–2791.
- [66] C.-C. Hsu, H.-T. Chen, W.-J. Tsai, and S.-Y. Lee, “Fast multi-view people localization using a torso-high reference plane,” in *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2017, pp. 1–4.
- [67] P. Peng, Y. Tian, Y. Wang, and T. Huang, “Multi-camera pedestrian detection with multi-view bayesian network model.” in *BMVC*, 2012, pp. 1–12.
- [68] Y. Yan, M. Xu, and J. S. Smith, “Multiview pedestrian localisation via a prime candidate chart based on occupancy likelihoods,” in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 2334–2338.
- [69] T. Klinger, F. Rottensteiner, and C. Heipke, “Probabilistic multi-person localisation and tracking in image sequences,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 127, pp. 73–88, 2017.
- [70] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, “Monocular 3d object detection for autonomous driving,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2147–2156.
- [71] T. Bagautdinov, A. Alahi, F. Fleuret, P. Fua, and S. Savarese, “Social scene understanding: End-to-end multi-person action localization and collective activity recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4315–4324.
- [72] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, “How far are we from solving pedestrian detection?” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1259–1267.

- [73] K. Kim, B. Heo, M. Byeon, and J. Y. Choi, “Deep learning architecture for pedestrian 3-d localization and tracking using multiple cameras,” in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 1147–1151.
- [74] T. Chavdarova and F. Fleuret, “Deep multi-camera people detection,” in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2017, pp. 848–853.
- [75] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, “3d object proposals for accurate object class detection,” in *Advances in Neural Information Processing Systems*. Citeseer, 2015, pp. 424–432.
- [76] B. Li, “3d fully convolutional network for vehicle detection in point cloud,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1513–1518.
- [77] B. Xu and Z. Chen, “Multi-level fusion based 3d object detection from monocular images,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2345–2353.
- [78] D. Xu, D. Anguelov, and A. Jain, “Pointfusion: Deep sensor fusion for 3d bounding box estimation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 244–253.
- [79] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, “Joint 3d proposal generation and object detection from view aggregation,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–8.
- [80] Z. Wang and K. Jia, “Frustum convnet: Sliding frustums to aggregate

- local point-wise features for amodal 3d object detection,” *arXiv preprint arXiv:1903.01864*, 2019.
- [81] S. Gerke, K. Muller, and R. Schafer, “Soccer jersey number recognition using convolutional neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 17–24.
- [82] Q. Ye, Q. Huang, S. Jiang, Y. Liu, and W. Gao, “Jersey number detection in sports video for athlete identification,” in *Visual Communications and Image Processing 2005*, vol. 5960. International Society for Optics and Photonics, 2005, p. 59604P.
- [83] G. Li, S. Xu, X. Liu, L. Li, and C. Wang, “Jersey number recognition with semi-supervised spatial transformer network,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1783–1790.
- [84] R. Zhang, C. Mu, M. Xu, L. Xu, and X. Xu, “Facial component-landmark detection with weakly-supervised lr-cnn,” *IEEE Access*, vol. 7, pp. 10 263–10 277, 2019.
- [85] M. Bertini, A. Del Bimbo, and W. Nunziati, “Matching faces with textual cues in soccer videos,” in *2006 IEEE International Conference on Multimedia and Expo*. IEEE, 2006, pp. 537–540.
- [86] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy, “Learning to track and identify players from broadcast sports videos,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 7, pp. 1704–1716, 2013.
- [87] T. Yamamoto, H. Kataoka, M. Hayashi, Y. Aoki, K. Oshima, and M. Tanabiki, “Multiple players tracking and identification using group detection and player number recognition in sports video,” in *IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2013, pp. 2442–2446.

- [88] A. Senocak, T.-H. Oh, J. Kim, and I. So Kweon, “Part-based player identification using deep convolutional representation and multi-scale pooling,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1732–1739.
- [89] S. Gerke, A. Linnemann, and K. Müller, “Soccer player recognition using spatial constellation features and jersey number recognition,” *Computer Vision and Image Understanding*, vol. 159, pp. 105–115, 2017.
- [90] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang, “Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 319–336, 2008.
- [91] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-cam: Visual explanations from deep networks via gradient-based localization,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.
- [92] L. Zheng, Y. Huang, H. Lu, and Y. Yang, “Pose-invariant embedding for deep person re-identification,” *IEEE Transactions on Image Processing*, vol. 28, no. 9, pp. 4500–4509, 2019.
- [93] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7291–7299.
- [94] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes challenge: A retrospective,” *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.

- [95] L. Bottou, “Large-scale machine learning with stochastic gradient descent,” in *Proceedings of COMPSTAT’2010*. Springer, 2010, pp. 177–186.