Multi-Camera 3D Position Estimation using Conditional Random Field

Shusuke Matsuda[1], Nattaon Techasarntikul[2], Hideyuki Shimonishi[2]
[1]Graduate School of Information Science and Technology, Osaka University
[2]Cybermedia Center, Osaka University
s-matsuda@ist.osaka-u.ac.jp, {techa.nattaon, shimonishi}.cmc@osaka-u.ac.jp

Abstract

In order to realize effective and safe human-robot collaboration where many humans and robots complement each other in close proximity, digital twin of the space would play a crucial role to monitor the behaviors of many robots and humans simultaneously and precisely in real time. Constructing such a digital twin requires estimating the precise 3D positions of instances in space, but Bluetooth sensors lack accuracy, and LiDARs are costly when covering wide areas. Therefore, we propose the use of multiple cameras to capture overlapping videos of the space and reconstruct the 3D positions of instances using geometrical methods. We propose a multimodal approach that utilizes not only vision features, but also position features, to detect the same objects in multiple cameras and use Conditional Random Field (CRF) to infer the identicality of objects detected in multiple cameras. The 3D positions of an instance taken from multiple 2D cameras are then geographically estimated. In the evaluation, we demonstrate the effects of CRF and multimodal approach, and achieve comparative performance with the state-of-the-art method.

1. Introduction

To address the social issues that will emerge in future societies, there would be a growing need for human-robot collaboration (HRC) [14], where humans and robots help each other in smart factories, warehouses, shop backyards, and so on. In such HRC, many humans and robots share a same space and collaborate in close proximity so that humans and robots complement each other to be more effective; sometimes humans assist robots' tasks, and sometimes vice versa. In order to realize both effectiveness and safety in such HRC, it would be crucial to monitor the behaviors of many robots and humans simultaneously and precisely in real time.

In recent years, there has been growing interest in a technology called "digital twin," which involves creating a virtual representation of the physical world. The purpose of a digital twin is to enable real time perception of the physical world, including the positions of humans and robots, using sensor devices and edge IoT systems. We believe that digital twin would play a critical role to realize HRC envisioned above. Behaviors of robots would be coordinated using information from the digital twin. Humans would be navigated to be more effective and safe to collaborate with the robots using the digital twin. To achieve this goal, lightweight and robust real-world perception, along with 3D position estimation of multiple objects, is essential.

While machine learning has made significant advancements in object recognition using images, achieving high recognition rates, there are challenges when considering broader digital twin applications. The power consumption and computational resources required for these applications are enormous. Additionally, addressing the decrease in recognition rates caused by information obtained from sensor devices is also an important challenge. The human brain serves as an example of a lightweight and robust system that effectively solves these problems.

Building upon these challenges, researchers have been actively exploring mathematical models to mimic the information processing mechanisms of the human brain. One such framework that has gained prominence is the "generative model." By employing probability distributions, generative models provide a means of accurately modeling the source of data generation. In this context, Bayesian estimation techniques are often employed, using multiple modalities. This approach enables the development of a lightweight and robust recognition system, facilitating the generation of the probabilistic digital twin.

On the other hand, an essential task in constructing this digital twin is estimating the 3D positions of objects in space. Several studies have investigated localization methods using depth and RGB cameras, as demonstrated in the works of [11, 20]. These methods have been shown to achieve higher localization accuracy compared to techniques that rely solely on radio wave positioning. However, the cost of depth sensors is typically high, and their depth accuracy range is limited. This presents a significant trade-

off between cost and accuracy in practical applications.

Therefore, in this study, we propose a method to estimate 3D positions by leveraging multiple RGB cameras to detect the same objects. One of the challenges in this work is accurately identifying and matching the same object across different camera views. Existing methods often focus on the similarity between two recognized objects and overlook the similarity between other objects in the scene. To address this challenge, we introduce a graph structure that considers the similarity between all recognized objects. This structure allows for efficient propagation of similarity information among different objects. However, traditional graph structures that consider the dependencies between nodes may not lead to a convergent solution. To overcome this, we incorporate Markov Random Field (MRF) into the graph structure. The MRF represents the conditional independence between the probability variables associated with the nodes and allows us to derive a Conditional Random Field (CRF). The CRF enables us to estimate the best combinations that represent the same objects. Based on the results obtained and the 2D positions captured by the multiple cameras, we perform a geographical estimation of the 3D positions. This estimation takes into account the positional and angular information of each camera and considers the directions in which recognized objects exist. This approach facilitates lightweight and robust estimation of 3D positions.

The rest of this paper is organized as follows. In Section 2, we provide a comprehensive review of related work in the field. Next, in Section 3, we introduce our novel approach to estimating 3D positions by detecting the same objects using the CRF framework. Following that, in Section 4, we conduct an evaluation of our proposed method. This paper concludes with Section 5.

2. Related Work

Vision-based Re-Identification (Re-Id) is a technique used to detect and identify the same objects across images captured by different cameras. It plays a critical role in object detection [1, 3, 4, 5]. These methods take advantage of deep learning to extract visual features, enabling high accuracy when trained and tested under consistent conditions. However, a drawback of these methods is that detection performance decreases when there are changes in conditions or background. In such cases, retraining becomes necessary to adapt the model to the new condition.

In contrast, an alternative and more general approach for detecting the same object using multiple cameras is to leverage the positions of recognized objects as features. Existing methods employ different strategies, such as projecting object positions onto the ground based on camera installation angles and positions, and utilizing spatial relationships to identify the same objects [13]. Another approach involves estimating the positions of recognized individuals using skeleton information, followed by object detection [10]. However, a limitation of these methods is that they assume that objects are in contact with a horizontal ground plane. This assumption poses difficulties in handling scenarios that involve steps or slopes. Therefore, in this paper, we propose a novel approach that not only utilizes visual features of recognized objects, but also incorporates their centroid positions as additional features.

Furthermore, distance learning methods have been widely proposed in the field of Re-ID [17, 18]. These methods employ Siamese networks and metric learning techniques to minimize the distance between the same objects and maximize the distance between different objects, leading to more accurate detection of the same objects. However, a limitation of these methods is that they only consider the similarity between pairs of objects and do not fully capture the similarity between other objects in the dataset.

Therefore, in this study, we propose the use of a graphical model to detect the same objects. Recently, there has been research focusing on Re-ID and object detection using graphical models, particularly methods employing CRF [2]. CRF allows for the representation of object states and their dependencies through nodes and edges, facilitating a more accurate classification of the entire set of objects. CRF has demonstrated its effectiveness in tasks such as sequence labeling and semantic segmentation, where considering contextual information is crucial. Similarly, in the context of detecting the same objects, the use of CRF proves beneficial, as it enables the incorporation of contextual cues for more accurate and reliable results [9].

Taking into account the aforementioned points, our study utilizes CRF for the detection of the same objects. We also take into account multimodal similarities by incorporating both vision and position features, thereby addressing the challenges that arise when relying on one type of feature.

3. Proposed Method

In this study, we define the source X as the state of existing actual instances in the real world, and the data Y as the vision and position feature vectors of the detected objects in each camera. To capture the true probability distribution of the real world, denoted $P_{true}(X)$, we construct a generative model $P_{model}(X)$ using MRF. Within this MRF construction, we use CRF as a learning model $P()$ to estimate the inferred results of the source data X based on the observed data Y. CRF is a method that seeks the optimal solution in a graphical representation by updating the states of the nodes. It is commonly used in tasks such as semantic segmentation and sequence labeling.

In the following section, we provide a detailed description of our 3D pose estimation method, as shown in Fig. 1. Our method involves constructing a graph based on the extracted features of objects captured by each camera. Sub-

sequently, we employ CRF to detect the same object within this graph. In CRF, unary and pairwise terms are calculated from the feature vectors, and inference is performed using belief propagation. Finally, we estimate the 3D positions based on the results obtained from the detection of the same objects.

3.1. Graph Definition

We present an overview of the graph in Fig. 2. In this study, we refer to entities existing in the 3D space as the "instance" denoted by I, and the objects recognized in the camera images as the local "object" denoted by L. Our aim is to model the state X of instances I_i and their 3D positions I^P. Therefore, we define $X = \{I_1^P, I_2^P, ..., I_{|I|}^P\}$.

These instances I give rise to the local observed objects L_l in each image captured by the respective camera. We consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the nodes $\mathcal{V}$ represent the local observed objects and are denoted as $\{L_1, L_2, ...L_{|L|}\}$. Each node L_l contains the feature values of the corresponding observed data, represented as $Y = \{Y_l | L_l \in \mathcal{V}\}$. The edges $\mathcal{E}$ are established between different nodes captured by different cameras, representing the connections between them. Additionally, we take into account the matching between the recognized objects L_l and the instances I_i. We express the probability distribution of these matches as hidden states denoted by $X' = \{L_l^i | L_l \in \mathcal{V}, i \in I\}$.

3.2. Conditional Random Fields

In our proposed method, we utilize CRF to estimate the hidden state X' and achieve the detection of the same objects based on the observed data Y. The conditional probability $P(X'|Y, X)$ in the CRF is expressed as:

$$P(X'|Y, X) = \frac{1}{Z} \exp\left(E(X'|Y, X)\right) \tag{1}$$

where $Z = \sum \exp\left(E(X'|Y, X)\right)$, and E denotes the Gibbs energy function:

$$E(X'|Y, X) = \sum_i \varphi_U(Y_l^P, X) + \sum_{L_l, L_k \in \mathcal{V}} \varphi_P(Y_l^P, Y_l^V, Y_k^P, Y_k^V, X) \tag{2}$$

where $\varphi_U and \varphi_P$ refer to the unary and pairwise terms, respectively, and Y^P and Y^V represent the position and vision feature vectors. The multiobject detection problem is transformed into an energy maximization problem, and the results $P(X')$ are then used to estimate the 3D positions $P(X_i)$. However, due to computational complexity, it is not feasible to calculate all combinations. Instead, we find the results of multi-object detection by maximizing the conditional probability:

$$x' = \underset{X'}{\operatorname{argmax}} P(X'|Y, X) \tag{3}$$

3.3. Belief Propagation

Belief propagation (BP) is a message passing algorithm that is used for inference in graphical models. It enables the exchange of information between neighboring nodes to update their states.

In the graph $\mathcal{G}$, we can derive the marginal distribution considering the conditional probabilities between neighboring nodes based on the Markov property. Hidden states X' are updated using messages that contain information from neighboring nodes. For example, the message m_{lk} from node L_l to L_k is expressed as:

$$m_{lk} = P(X_l'|Y_l, X_i)\varphi_P(Y_l^P, Y_l^V, Y_k^P, Y_k^V, X) \tag{4}$$

The messages are then used to update the states of nodes $P(X_l'|Y_l, X_i)$, resulting in:

$$P(X_l'|Y_l, X_i)^* = P(X_l'|Y_l, X_i) + \sum_{k \sim l} m_{kl} \tag{5}$$

where $\varphi_U(Y_l^P, X)$ represents the initial states of $P(X_l'|Y_l, X_i)$, and $k \sim l$ denotes that node L_k is adjacent to node L_l. By repeatedly performing these updates, we efficiently obtain the optimal posterior probability $P(X'|Y, X)$ for each node.

3.4. Unary Term

The unary term in Eq. 2 derives the hidden state X' based on the observed position feature Y^P and the position of the instance I^P. To obtain the candidates of Y^P, we perform hierarchical clustering and calculate the Euclidean distance between these candidates. Each detected object is then considered for similarity in the unary term.

First, we perform a coordinate transformation using camera parameters to obtain the candidates for Y^P. RGB cameras typically undergo a calibration process to determine lens distortion, focal length, camera installation position coordinates, and angles. With these parameters, points on the image plane can be projected as lines in 3D space. In this method, these lines represent the position feature vectors $Y^P = (Y_1^P, Y_2^P, ...Y_{|L|}^P)$. Fig. 3 provides a top-down view that illustrates these position features, with the red lines indicating Y^P. Among these position features, the lines captured by different cameras converge most closely at the common perpendicular point. It is assumed that positions where such convergence occurs frequently are likely to correspond to instances.

Consequently, we perform hierarchical clustering analysis at the midpoint of this common perpendicular point to detect the same object and estimate its 3D position. Hierarchical clustering is suitable for this method, as it does not re-

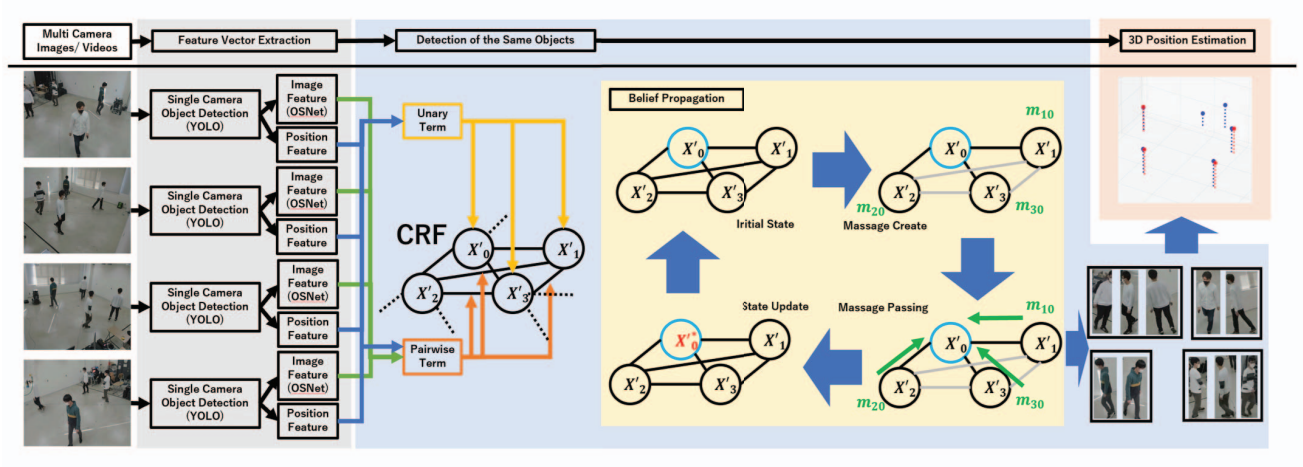

Figure 1: Method overview

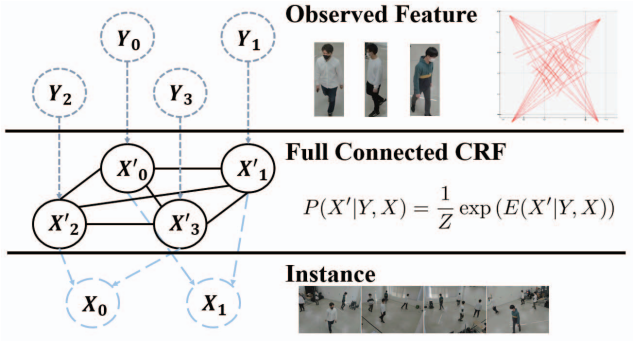

Figure 2: Graph definition

quire specifying the number of clusters beforehand. Instead, a threshold is set as the acceptable distance for position estimation errors. The center coordinates of these clusters are defined as the position features of the instance, denoted as $I_i^P (1 \leq i \leq |I|)$, and the Euclidean distance d between the position feature Y_l^P of the detected object L_l is considered. In this approach, we use the probability density of d in a Gaussian distribution with a mean of 0 and a variance of T. This T represents the threshold used in hierarchical clustering. Thus, the probability distribution $P(X'|Y_l^P, I^P)$ of the detected object L_l in clustering can be expressed as:

$$P(X'|Y_l^P, I^P) = Gaussian(\sqrt{(I^P - Y_l^P)^2}, 0, T) \quad (6)$$

where $Gaussian()$ represents the first-order Gaussian distribution, and the similarity of the unary term $\varphi_U(Y_l^P, X)$ is expressed as:

$$\varphi_U(Y_l^P, X) = P(X'|Y_l^P, I^P) \quad (7)$$

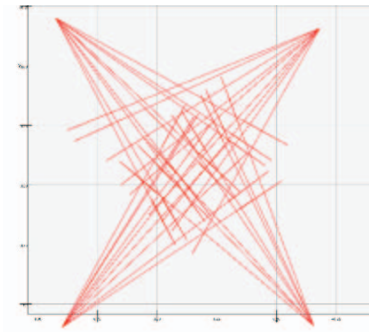

Figure 3: Positional feature vector form cameras to objects

3.5. Pairwise Term

The result of position clustering alone may lead to the convergence of straight lines, which are positional features, at unintended coordinates. Moreover, considering only the similarity between objects and detected entities does not capture the similarity between the detected entities themselves. To address these issues, we incorporate probabilities of vision similarity and positional similarity between detected entities into pairwise terms. By using feature vectors Y^V extracted using an existing CNN model and calculating the cosine similarity, as well as considering the positional characteristics Y^P and computing the Euclidean distance, we define the pairwise term $\varphi_P(Y_l^P, Y_l^V, Y_k^P, Y_k^V, X)$ as follows:

$$\varphi_P(Y_l^P, Y_l^V, Y_k^P, Y_k^V, X) = cos(Y_l^V, Y_k^V) dis(Y_l^P, Y_k^P) \quad (8)$$

$$dis(Y_l^P, Y_k^P) = T - euclid(Y_l^P, Y_k^P) \quad (9)$$

where $cos()$ represents the cosine similarity and $euclid()$ denotes the Euclidean distance.

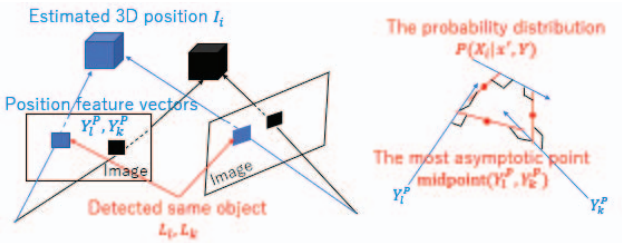

Figure 4: 3D position estimation from multi-view images

3.6. 3D Position Estimation

In the proposed method, the 3D position of each instance I_i is estimated based on the results of the detection of the same objects $P(x')$, as depicted in Fig. 4. For every pair of the same objects detected L_l and L_k from different cameras, we consider the probability distribution $P(X_i)$, where each element represents the points at which their position feature vectors Y_l^P, Y_k^P are most convergent. The 3D position I_i is obtained as a weighted average of the probability distribution $P(X_i)$, and the "most convergent" point is defined as the midpoint of the common perpendicular lines $Y_l^P and Y_k^P$. We denote this midpoint by $midpoint(Y_l^P, Y_k^P)$.

$$P(X_i|x', Y) = (midpoint(Y_l^P, Y_k^P)|l, k \in x_i') \quad (10)$$

$$I_i = \frac{1}{z} \sum_{l,k \in x_i'} midpoint(Y_l^P, Y_k^P) \\ (P(x'|Y_l, X) + P(x'|Y_k, X)) \quad (11)$$

$$z = \sum_{l,k \in x_i'} (P(x'|Y_l, X) + P(x'|Y_k, X)) \quad (12)$$

4. Experiments

We evaluate the performance of the proposed method in a multi-camera 3D position estimation scenario. For the evaluation, we use both original data for conducting ablation studies and an open dataset for comparison with an existing method [12]. The proposed method aims to detect the same objects across multiple cameras by leveraging both positional and vision features and utilizing a graph-based approach.

4.1. Environment Setup

In this evaluation, we have implemented the proposed method using Python. We perform object detection in each camera using YOLO v5 [15, 8]. To extract the vision feature vectors used in pairwise terms, we utilize OSNet [21]. For the evaluation, we target video input, where we obtain object detection results using YOLO and gather RGB and position information from each frame of the video. The RGB information is transformed into a 512-dimensional image feature vector using OSNet, while the position information is converted into a position feature vector through coordinate transformation using camera parameters. By obtaining these input information for each frame, we acquire the necessary data to construct the graph for further processing.

4.2. Dataset and Evaluation Metrics

The purpose of this study is to address the challenge of estimating 3D positions using multiple cameras, which is not a commonly explored approach. Consequently, there is a scarcity of available datasets that provide accurate 3D position coordinates, which are essential for evaluating such methods. To conduct the ablation study, we created our own dataset specifically for this purpose. In our original dataset, we set up cameras at the corners of an indoor space measuring approximately 6 by 7 meters, positioned at a height of 2.5 meters, as depicted in Fig. 7. We recorded video data at a frame rate of 30 fps for approximately 1 minute. During the video recording, 4 individuals engaged in random walks for the first 30 seconds. Subsequently, 2 individuals exited the scene, while the remaining two individuals continued random walking for an additional 30 seconds.

For our original dataset, we manually annotated the ground truth data, which includes object IDs and 3D positions. First, we fixed the correct 3D position data based on the height of each individual for the vertical component. Next, we visually confirmed the grid of 10 cm intervals superimposed on the image of each frame. It is important to note that due to the manual annotation process and the limitations of visual estimation, the ground-truth data itself may have an error of approximately 10 cm.

To compare our proposed method with an existing approach, we utilized the EPFL Multi-camera pedestrians video Terrace dataset [6]. This dataset consists of video footage captured by four cameras, featuring nine individuals walking in a random manner. The EPFL dataset provides ground truth data for the positions of individuals, which are expressed as correct positions in the ground plane. The ground-truth positions are provided in a grid format with a resolution of 25 cm by 25 cm.

Furthermore, in evaluating the performance of object detection, we employ several evaluation metrics: Homogeneity (H), Completeness (C), V-Measure (V) [16], Adjusted Rand Index (ARI) [7], and Adjusted Mutual Information (AMI) [19]. Homogeneity measures the extent to which each cluster contains only samples of the same class, while Completeness evaluates whether all samples of a given class are assigned to the same cluster. The V-measure is the harmonic mean of H and C.

These three metrics assess the clustering results, but may not be sufficient in cases of random labeling. Therefore, we also utilize ARI and AMI, which focus on the similarity of assignments or mutual information. ARI quantifies

(a) Camera 0 **(b)** Camera 1 **(c)** Camera 2 **(d)** Camera 3

Figure 5: Terrace dataset

(a) Camera 0 **(b)** Camera 1 **(c)** Camera 2 **(d)** Camera 3

Figure 6: Original video data

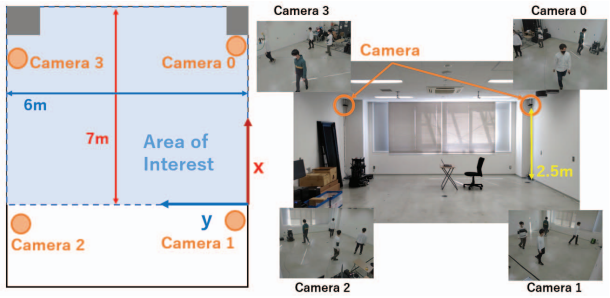

Figure 7: The setting of experimental space

whether two elements are grouped in the same cluster, while AMI incorporates mutual information to assess the clustering agreement.

Each of these metrics produces a score ranging from 0 to 1, with 1 indicating perfect clustering. However, for ease of understanding, we present scores in the range of $[0, 100]$.

In addition to these clustering metrics, we evaluate the performance of our proposed method for 3D position estimation by considering the distribution of errors in relation to the ground-truth positions. This evaluation metric provides insight into the accuracy and precision of our method for estimating 3D objects' positions.

4.3. Parameter Setup

In this method, BP is used for detecting the same objects, and the frequency of graph updates must be specified as a parameter. To determine the appropriate number of iterations for BP, we conducted an evaluation and fixed the parameters accordingly. Fig. 8 illustrates the results of the evaluation, showing the AMI, average 3D position error, and standard deviation of 3D position error for different numbers of iterations in the range of 0 to 10. The results for 0 iterations represent the performance based solely on the unary term. Based on these results, we observe a significant improvement in all metrics with the introduction of BP. The metrics stabilize after around two iterations. In our proposed method, all nodes in the CRF are connected within a 2-hop range. Therefore, with two iterations of BP, each node incorporates the states of all other nodes, leading to improved performance and accuracy in detecting the same objects. In the following evaluation, we set the number of iterations to two.

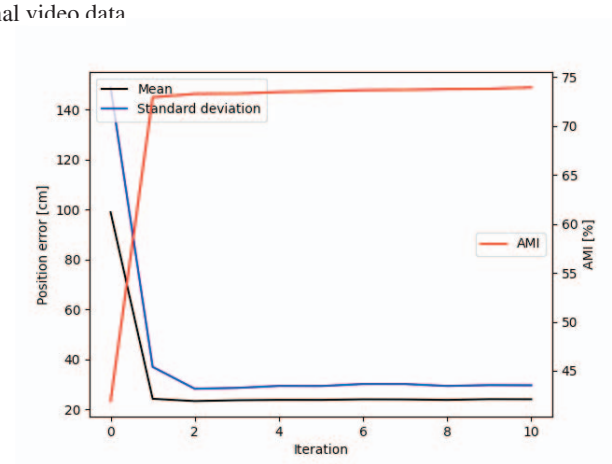

Figure 8: Evaluation of the number of iterations

4.4. Evaluation of Same Object Detection

In this section, we evaluate the results of the same object detection. We begin by assessing the impact of CRF and the multimodal approach using our original dataset. Next, we compare the performance of our proposed method

Table 1: Result of detection of same objects in original dataset[%]

	ARI	AMI	H	C	V
Unary	45.12	51.13	98.35	89.51	93.72
CRF & p	**78.86**	**81.46**	99.57	**94.80**	**97.12**
CRF & p + v	77.08	80.60	**99.72**	94.51	97.04

Table 2: Result of detection of same objects in Terrace dataset [%]

	ARI	AMI	H	C	V
CRF & p + v	82.97	84.01	**99.64**	**97.63**	**98.62**
GNN-CCA [12]	**83.07**	**86.77**	93.59	92.03	92.66

with a state-of-the-art GNN-CCA [12]. The evaluation aims to achieve the detection of the same objects under conditions similar to our study. The GNN-CCA method also utilizes similarities between images and positions, employing a graphical model for the detection task.

4.4.1 Original Dataset

In the proposed method, we employ CRF along with multiple features to achieve a 3D position estimation. To assess the contribution of these components to the accuracy of 3D position estimation, we conducted an ablation study comparing three scenarios: "Unary term only," "CRF with positional feature," and "CRF with both position and vision features." Throughout this section, we will refer to these scenarios as "Unary," "CRF + Position," and "CRF + Position + Vision," respectively.

First, we assess the performance of the same object detection task by comparing the results obtained under three different scenarios using our original video dataset. We use several evaluation metrics including ARI, AMI, H, C, and V. The results of this evaluation are presented in Table 1.

The results indicate that the inclusion of CRF significantly enhances the performance compared to the situation with the unary term only. However, the incorporation of vision similarity does not contribute significantly to the overall improvement in performance.

4.4.2 Terrace Dataset

To evaluate the performance of our proposed method, we conducted a comparison with the state-of-the-art approach using the EPFL Multi-camera pedestrians video Terrace dataset. The state-of-the-art method chosen for comparison is GNN-CCA [12]. The comparison results between our proposed method and GNN-CCA are presented in Table 2. These results demonstrate that our proposed method achieves comparable performance in terms of detecting the same objects compared to the existing state-of-the-art approach, GNN-CCA.

Moreover, it is worth noting that GNN-CCA requires training to determine the weights of the constructed graphs and also involves post-processing steps to improve the accuracy of the detection. On the other hand, our proposed method does not require graph training and does not involve any post-processing. This makes our method computationally lighter and more straightforward to implement compared to the existing approach. Our proposed method achieves comparable performance in detecting the same objects.

Table 3: Result of 2D position estimation in Terrace dataset [cm]

	error Ave.	error S.D.	error Max
CRF & p + v	33	75	846

Table 4: Result of 3D position estimation in original dataset [cm]

	error Ave.	error S.D.	error Max
Unary	88	135	935
CRF & p	31	43	366
CRF & p + v	**24**	**30**	**317**

4.5. Evaluation of Position Estimation

Next, we evaluate the accuracy of the 3D position estimation using both the Terrace dataset and our original dataset, which provides ground-truth data for both 2D and 3D positions.

4.5.1 2D Position Estimation

We evaluate the accuracy of the proposed method's position estimation using ground truth data. For evaluation, we calculate the average error, standard deviation, and maximum error in position estimation. In the Terrace dataset, the ground truth data provide correct positions on the ground plane for individuals and are provided in a grid format with a resolution of 25 cm by 25 cm. We consider the center of each grid to be the ground truth position for evaluation, acknowledging that the ground truth data itself have an error of about 10 cm. The results of the evaluation are presented in Table 3. Based on the table, the proposed method achieves a position estimation within an error of 20 cm for the 2D position.

4.5.2 3D Position Estimation

We evaluate the 3D position estimation using the provided 3D ground-truth position data. The results of the evaluation are shown in Table 4.

The results indicate that the use of CRF significantly improves the performance compared to the Unary Term only situation in terms of detecting the same objects. Additionally, incorporating the vision feature leads to further improvement in the accuracy of 3D position estimation. This improvement can be attributed to the relationship between the evaluation metric H (as shown in Fig. 1) and the estimation of the 3D position. The metric H is associated with the

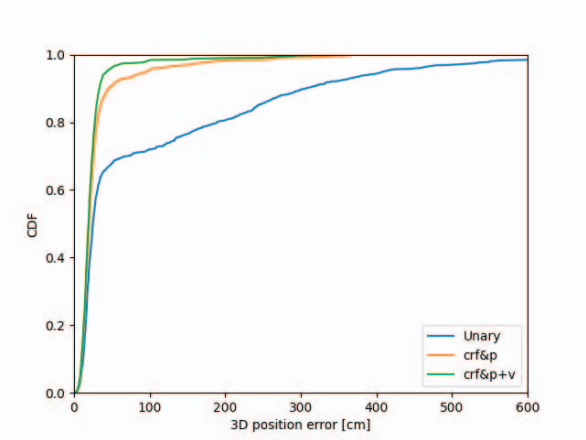

Figure 9: 3D position error comparison

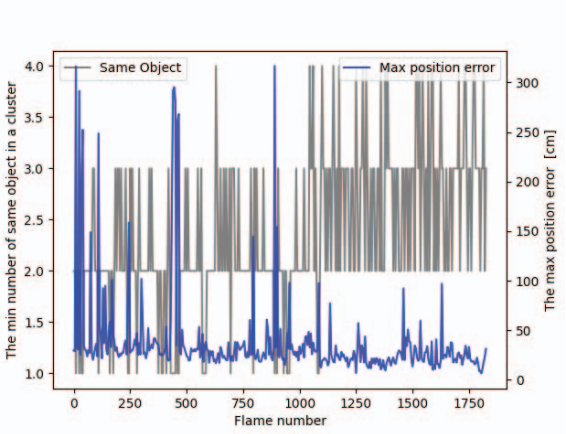

Figure 10: The max position error and the min number of same objects in a cluster per frame

number of clusters consisting of identical objects. In our 3D position estimation method, if different objects are mistakenly identified as identical objects, it can lead to increased positional errors. Therefore, the use of multiple features facilitates the effective detection of the same objects, which is beneficial for accurate 3D position estimation.

To analyze the error in 3D position estimation, we consider the cumulative distribution function (CDF) and present the results in Fig. 9. From these results, it is evident that the "CRF" methods, especially "CRF & p + v," significantly improve the position estimation error. The "CRF & p + v" method shows a smaller variance in errors and achieves an improved position estimation by improving the accuracy of detecting the same objects, as demonstrated by the improved precision in the detection of the same objects. Furthermore, the distribution that was previously present around 100 cm in the "CRF & p" method is improved in the "CRF & p + v" method due to the incorporation of vision features. These findings indicate that the use of multiple features, particularly the combination of two features, leads to a more accurate and reliable 3D position estimation. Therefore, it can be confirmed from this Fig. 9 that utilizing multiple features enables the detection of the same object that is suitable for 3D position estimation.

Finally, we examine the relationship between the detection of the same objects and the error in 3D position estimation. We compare the maximum 3D position error with the minimum number of objects representing the same instance in a cluster for each frame, using the "CRF & p + v" method. Fig. 10 illustrates the relationship between the detection of the same objects and the maximum 3D position error. The dataset used in this evaluation consists of frames where the number of people decreases from 4 to 2 around the 1000th frame. After the 1000th frame, there is an improvement in both the maximum position error and the number of the same objects detected. As the number of detections of the same object decreases, the position error tends to increase. In particular, frames with only one same object exhibit a significantly large error, exceeding 100 cm. Conversely, in situations where more same objects are detected, there is a noticeable reduction in positional error. When all four identical objects, which correspond to the number of cameras, can be detected, the error is suppressed to around 20 cm. On the basis of these observations, it can be inferred that detecting more identical objects and capturing objects with a higher number of cameras contribute to a reduction in the error in 3D position estimation.

4.5.3 Inference Time

In addition, we need to discuss about the inference time of the program used above. We run the python program on Intel i7 with a 8-cores CPU running at 3.60GHz. In this situation, it is capable of execution at approximately 10 fps, depending also on the number of detected objects. From this result, the proposed method is capable of estimating positions in real time and can be utilized for HRC.

5. Conclusion

We propose a method for 3D position estimation to realize human-robot collaboration. Our approach utilizes a graph structure with a MRF as a generative model and a CRF as a learning model for detecting the same objects. We introduce a multimodal approach that incorporates both vision and positional characteristics. In the evaluation, we conduct an ablation study to demonstrate the effectiveness of BP in our proposed method. Furthermore, through comparative evaluation using open datasets, we achieve performance comparable to the state-of-the-art method for detecting the same objects. Additionally, we demonstrate the ability to estimate 3D positions with an error of approximately 20 cm.

Acknowledgement

These research results were partly obtained from the commissioned research (No.00701) by National Institute of Information and Communications Technology (NICT), Japan.

References

[1] Ejaz Ahmed, Michael Jones, and Tim K Marks. An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3908–3916, 2015.

[2] Dapeng Chen, Dan Xu, Hongsheng Li, Nicu Sebe, and Xiaogang Wang. Group consistent similarity learning via deep crf for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8649–8658, 2018.

[3] Dapeng Chen, Zejian Yuan, Gang Hua, Nanning Zheng, and Jingdong Wang. Similarity learning on an explicit polynomial kernel feature map for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1565–1573, 2015.

[4] Dapeng Chen, Zejian Yuan, Jingdong Wang, Badong Chen, Gang Hua, and Nanning Zheng. Exemplar-guided similarity learning on polynomial kernel feature map for person re-identification. *International Journal of Computer Vision*, 123(3):392–414, 2017.

[5] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 403–412, 2017.

[6] Francois Fleuret, Jerome Berclaz, Richard Lengagne, and Pascal Fua. Multicamera people tracking with a probabilistic occupancy map. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):267–282, 2007.

[7] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2:193–218, 1985.

[8] Glenn Jocher. Yolov5 by ultralytics, 2020.

[9] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24: , 2011.

[10] Joao Paulo Lima, Rafael Roberto, Lucas Figueiredo, Francisco Simoes, and Veronica Teichrieb. Generalizable multi-camera 3d pedestrian detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1232–1240, 2021.

[11] Jianan Liu, Liping Bai, Yuxuan Xia, Tao Huang, Bing Zhu, and Qing-Long Han. Gnn-pmb: A simple but effective online 3d multi-object tracker without bells and whistles. *IEEE Transactions on Intelligent Vehicles*, page , 2022.

[12] Elena Luna, Juan C SanMiguel, José M Martínez, and Pablo Carballeira. Graph neural networks for cross-camera data association. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[13] Peixi Peng, Yonghong Tian, Yaowei Wang, Jia Li, and Tiejun Huang. Robust multiple cameras pedestrian detection with multi-view bayesian network. *Pattern Recognition*, 48(5):1760–1772, 2015.

[14] Silvia Proia, Raffaele Carli, Graziana Cavone, and Mariagrazia Dotoli. Control techniques for safe, ergonomic, and efficient human-robot collaboration in the digital industry: A survey. *IEEE Transactions on Automation Science and Engineering*, 19(3):1798–1819, 2021.

[15] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788. IEEE Computer Society, 2016.

[16] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 410–420, 2007.

[17] Rahul Rama Varior, Mrinal Haloi, and Gang Wang. Gated siamese convolutional neural network architecture for human re-identification. In *European conference on computer vision*, pages 791–808. Springer, 2016.

[18] Rahul Rama Varior, Bing Shuai, Jiwen Lu, Dong Xu, and Gang Wang. A siamese long short-term memory architecture for human re-identification. In *European conference on computer vision*, pages 135–153. Springer, 2016.

[19] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research*, 11:2837–2854, 2010.

[20] Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10359–10366. IEEE, 2020.

[21] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Omni-scale feature learning for person re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3702–3712, 2019.