RC6D: An RFID and CV Fusion System for Real-time 6D Object Pose Estimation

Bojun Zhang*†¶, Mengning Li‡¶, Xin Xie§, Luoyi Fu‡, Xinyu Tong*†, Xiulong Liu*†
*College of Intelligence and Computing, Tianjin University, China
†Tianjin Key Laboratory of Advanced Networking (TANK)
‡School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, China
§The Hong Kong Polytechnic University, China
¶: Co-primary authors

***Abstract*—This paper studies the problem of 6D pose estimation, which is practically important in various application scenarios such as robotic-based object grasping, obstacle avoidance in autonomous driving scene, and object integration in mixed reality. However, existing methods suffer from at least one of the five major limitations: dependence on object identification, complex deployment, difficulty in data collection, low accuracy, and incomplete estimation. To overcome the above limitations, this paper proposes an RC6D system, which is the first to estimate 6D poses by fusing RFID and Computer Vision (CV) data with multi-modal deep learning techniques. In RC6D, we first detect 2D keypoints through a deep learning approach. We then propose a novel RFID-CV fusion neural network to predict the depth of the scene, and use the estimated depth information to expand the 2D keypoints to 3D keypoints. Finally, we model the coordinate correspondences between the detected 2D-3D keypoints, which is applied to estimate the 6D pose of the target object. When implementing RC6D, we mainly address the following three technical challenges. (i) To predict 6D poses without using the CAD model, we propose a network architecture for monocular depth estimation. (ii) To train the neural network for 6D pose estimation without time-consuming 6D labeling, we use an unsupervised learning algorithm based on 2D-3D point pair matching. (iii) To detect the subject of the object without identification, we leverage optical flow to restrict the object and RFID to directly obtain its information. The experimental results show that the localization error of RC6D is less than $10\,\mathrm{cm}$ with a probability higher than 90.64% and its orientation estimation error is less than 10° with a probability higher than 79.63%. Hence, the proposed RC6D system performs much better than the state-of-the-art related solutions.**

***Index Terms*—6D Pose, RFID, Camera, Multi-modal Fusion.**

I. INTRODUCTION

A. Motivation and Problem Definition

The widespread use of Internet-of-Things (IoT) techniques allows us to better learn and control everything in physical world. Identifying location and orientation of objects will benefit various IoT-enabled applications, such as robotic-based object grasping [1], obstacle avoidance in autonomous driving [2], and object integration in mixed reality rendering [3]. For example, in a factory, items misplaced upside down can be automatically identified and corrected using robotic arms; in the context of automatic driving, the car can predict the 6D pose of obstacles on the road, and perform the corresponding obstacle avoidance operations; in mixed reality applications, the estimated 6D pose of objects can be integrated into virtual environment, thereby offering human an immersion experience. The problem of 6D pose estimation is formally defined as follows. As illustrated in Fig. 1, given a universal coordinate system X-Y-Z originating from point O, each object also has its own coordinate system X'-Y'-Z' originating from point O'. Estimation of object's 6D pose is to estimate the absolute location of O' in the universal coordinate system and angle differences α=$\angle X''O'X'$, β=$\angle Y''O'Y'$, γ=$\angle Z''O'Z'$. Note that, the three axes of the coordinate system X''-Y''-Z'' are parallel to those of the universal system, respectively.

Correspondence author: Xiulong Liu.

B. Limitations of Prior Art

The closely related existing methods suffer from at least one of the following five major limitations. **Relying on object identification:** The accuracy of 6D pose estimation systems [4]–[6] significantly depends on the object detection accuracy. However, the viewpoint variation, deformation and illumination conditions may significantly lower the object identification accuracy, which will inevitably decrease the accuracy of 6D pose estimation. **Complex deployment:** Multiple types of devices need to be simultaneously deployed, *e.g.*, five devices including Radar, GPS, IMU, GNSS and camera are used in [7], which incurs poor system deployability [7], [8]. **Difficulty in data collection:** It is difficult to obtain the point cloud of the object because most of the methods rely on multi-view data for 3D reconstruction, which puts a heavy burden on the information collection process [2], [5]. **Low accuracy:** Existing category-level methods [1], [9]–[11] suffer from low accuracy of 6D pose estimation, since they apply a CAD approximation for pose estimation. **Incomplete estimation:** Different types of collected data cannot be synchronized, which means that a registration is needed, though the accuracy is usually unsatisfactory [12].

C. RC6D System in a Nutshell

To overcome the above limitations, this paper proposes the RFID-CV fusion based 6D pose estimation (RC6D) system. As illustrated in Fig. 1, the RC6D system mainly consists of three types of hardware devices: (i) an RFID reader with an antenna; (ii) an Azure Kinect DK with a depth sensor and a RGB camera; (iii) a backend server. The RC6D system mainly consists

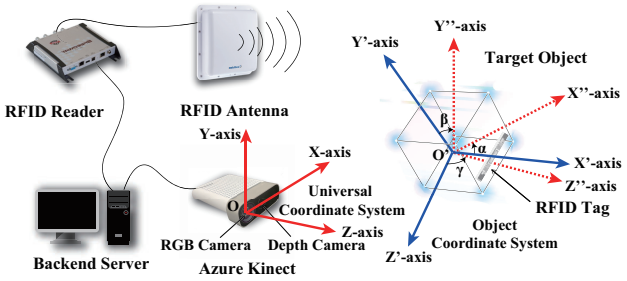


Fig. 1. Illustrating 6D pose estimation of target object

of three major blocks: 2D keypoint detection, 3D keypoint detection, and 6D pose estimation. Specifically, to detect 2D keypoints, we collect and manually label an image dataset for training and testing, and conduct extensive experiments with different object types, locations and orientations. An encoder-decoder model is trained on the above dataset to detect the 2D keypoints in the given RGB image. To detect 3D keypoints, we convert the problem of detecting 3D keypoint coordinates into a problem of generating depth maps. We propose a multi-modal data fusion algorithm to incorporate image and RFID data (*e.g.*, phase, RSSI, and doppler values), to obtain depth information about the object surface, whose ground-truth depth value can be automatically captured by the depth camera of the Kinect. Finally, we apply the Perspective-n-Point (PnP) algorithm to solve the 6d pose of the object.

D. Challenges and Solutions

We need to solve the following three technical challenges when implementing our RC6D system.

The first challenge is how to estimate object pose without scene depth and computer-aided design (CAD) model. Existing CV-based pose estimation methods heavily depend on scene depth or CAD models, so it is difficult to deploy them in practice. This paper proposes an RFID-CV fusion solution. RFID has the advantages of fast classification and coarse-grained localization, while it cannot provide pixel-level accuracy of object details. While RGB images can provide considerable texture and color features, they face challenges in target classification and localization. We design a multi-modal fusion scheme to integrate RFID and CV, which are complementary to each other, for achieving high accurate scene depth reconstruction.

The second challenge is how to reduce the considerable manpower consumption during the data labeling processes. 2D image labeling is an acknowledged necessary process, which has to be done artificially. In contrast, existing 3D image labeling is so complicated that it generally relies on auxiliary tools (*e.g.*, AprilTag [13]) and takes up a lot of the labeling time. Thus, there is still significant room for reductions in manpower consumption. This paper leverages unsupervised learning to automatically estimate 6D pose, and avoids any undesired significant changes in 3D caused by the slight changes in 2D.

The third challenge is how to make the RC6D system more robust for estimating the pose of an object. Due to the high requirements of object detection, it is of great importance

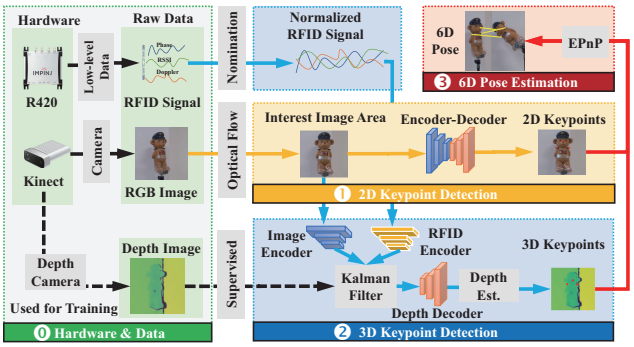


Fig. 2. System overview

to maintain a robust object detection model working under various conditions. However, the common object detection is sensitive to the environment, which lowers the accuracy of pose estimation. To address this issue, we use RFID to directly identify the subject of an object. The major advantage is that every RFID tag has a unique EPC that exactly identifies the tagged item, thereby completely avoiding detection errors.

E. Novelty and Advantages over Prior Art

This paper for the first time addresses the problem of estimating 6D pose of tagged object without prior knowledge of the 3D models. The technical novelty of our work lies in tackling three key challenges: estimating the object pose without knowing its 3D model, labeling objects through deep learning, and ensuring that the proposed system is robust under various conditions. The RC6D system has three main advantages over the previous works: (i) Compared with [9] [14] [15], RC6D estimates 6D poses without the need for accurate CAD models of objects. (ii) Compared with [16] [17] [8] that requires complex equipment deployment, we implement the system with an RFID antenna, an RFID reader, a camera and a tag attached to the target object. (iii) Unlike the state-of-the-art methods that pay many efforts on data labeling, RC6D leverages unsupervised learning to make our system suitable for real applications and easy for transfer learning. The experimental results show that localization error of RC6D is less than $10\,\mathrm{cm}$ with a probability higher than 90.64% and its average orientation estimation error is less than 10° with a probability higher than 79.63%.

The remainder of this paper is organized as follows. We present the preliminary knowledge in Section II, and detailed system design in Section III. Section IV presents the implementation and the experimental results. We discuss the related work in Section V and conclude the paper in Section VI.

II. Preliminary Knowledge

This section will present some preliminaries of the devices, data and basic mechanisms in our RC6D system.

A. RFID

Typically, an RFID system consists of four major components: a reader, a scanning antenna, tags, and a data processing system. The reader communicates with the tag using a

backscatter radio link. Specifically, the electromagnetic wave from the reader's scanning antenna induces a voltage on the tags, causing the tag to gain power and modulate its data onto the backscatter signal and then transmit it back to the reader. In addition to the tags' EPC data, the reader can also obtain the low-level signal data that implies the spatial relationships between the tags and the reader: Radio Frequency (RF) phase value, Received Signal Strength Indication (RSSI), and the received signal doppler value.

RF Phase: The phase is often used in the field of RFID related data mining [18]. The RF phase value is a basic attribute of RF carrier waves that describes the degree to which the received signal is offset from the sent signal, ranging from 0 to 2π. Since a complete RFID signal transmission process is a round trip, the total distance traversed by the signal travels is $2d$. In addition to the RF phase rotation over distance, the reader's transmit circuits, the tag's reflection characteristics, and the reader's receiver circuits will all introduce some additional phase rotation ϕ_T, ϕ_{reader}, and ϕ_{tag}, respectively. The total phase rotation can be expressed as:

$$\phi = \left[\frac{2d}{\lambda} \cdot 2\pi + \phi_T + \phi_{\text{reader}} + \phi_{\text{tag}}\right] \mod 2\pi, \quad (1)$$

where λ is the wavelength and $\mu = \theta_T + \theta_{\text{reader}} + \theta_{\text{tag}}$ is a constant value determined by the hardware characteristics. Most commercial RFID readers (*e.g.*, as ImpinJ R420) are able to report θ as the difference between the transmitted and the received RF carriers, which reflects the distance d between the tag and the antenna. However, as the phase is a periodic value that repeats every λ in the distance of signal propagation, we cannot directly use these phase values to pinpoint deterministic tag locations.

RSSI: RSSI measures the power of the received radio signal, and it has a negative logarithmic relationship with the distance between the tag and the reader. Specifically, the power of signal received (P_R) by an RFID reader can be expressed as:

$$P_R = \frac{G_T^2 \cdot \lambda^2 \cdot \sigma}{(4\pi)^3 \cdot R^4} \cdot P_T, \quad (2)$$

where P_T and G_t are the transmission power and the antenna gain, respectively. However, multipath propagation and undesired environmental interference can combine with the primary backscatter, thereby increasing or decreasing the received signal power at the reader receiver. Thus, compared with tag phase, RSSI is an unreliable indicator and seldom used in the recent wireless sensing solutions.

B. Camera

We use Kinect to capture images in our RC6D system, which is equipped with a 12 megapixel RGB camera as well as a megapixel-depth camera. The depth camera implements the Amplitude Modulated Continuous Wave (AMCW) Time-of-Flight (ToF) principle. The camera casts modulated illumination in the near-IR (NIR) spectrum onto the scene. These measurements are processed to generate a depth map. A depth map is a set of z-coordinate values for every pixel of the image,

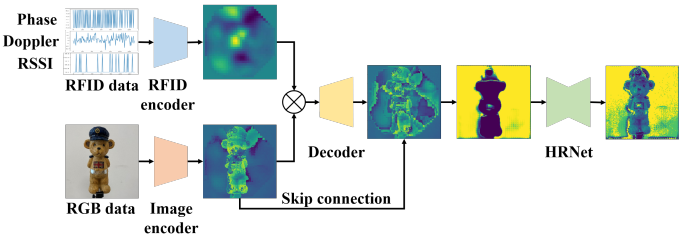


Fig. 3. Architecture for depth reconstruction with CV-RFID fusion

measured in units of millimeters. Based on RGB image stream, we obtain the motion information of the object between two adjacent frames through the optical flow algorithm, which leverages the changes of pixels in the time domain and the correlation between adjacent frames to find the correspondence between the frames. Specifically, we assume that the brightness is constant and the movement of the object is continuous with the time. Considering a 2D+t dimensional case, the intensity of a pixel at (x, y, t) will move Δx, Δy and Δt between two adjacent frames, and the brightness constancy constraint is:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t). \quad (3)$$

Assuming Changes in time do not cause drastic changes in location, the optical flow constraint equation at $I(x, y, t)$ is

$$\frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} = 0, \quad (4)$$

where V_x and V_y are respectively the x and y components of the optical flow $I(x, y, t)$; $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ are the derivatives of the image at (x, y, t) in the corresponding directions, which can be calculated from the information of the image. Solving this equation, we can obtain the optical flow vector (V_x, V_y).

III. Detailed Design of The RC6D System

In this section, we first present the overview of the RC6D system, then explain the building blocks in detail sequentially.

A. System Overview

As illustrated in Fig. 2, the RC6D system mainly consists of three blocks: 2D keypoint detection, 3D keypoint detection and 6D pose estimation. First, in the block of 2D keypoint detection, we first record the RGB frames with the camera and roughly localize the moving objects within a fixed size 2D window using the Lucas-Kanade (L-K) based optical flow. We then feed the cropped image within the window into a convolutional-based encoder-decoder neural network to obtain a heatmap showing the pixel-level probability distribution of the individual keypoints. Next, in the block of 3D keypoint detection, we obtain the RFID signal from the reader, and integrate the data with the RGB frame to reconstruct the depth map of the monitoring area using a multi-modal neural network. The reconstructed depth map should be aligned with the RGB frame pixel by pixel, which can be used to obtain the depth information for each 2D keypoint in the RGB image. Finally, in the block of 6D pose estimation, we estimate the location and orientation of the object by matching the pairs of 2D and 3D keypoints via the EPnP algorithm. We will

elaborate on the technical details of the above building blocks in the following sections.

B. 2D Keypoint Detection

First, we would like to predict the 2D pose of the object in a given RGB image, where the task is formulated as keypoint detection. Specifically, we transform the keypoint detection problem into an image generation problem, and address this problem by using a convolutional based encoder-decoder that generates 2D keypoint heatmaps.

1) Target Object Detection: Given a series of RGB image frames captured by the camera, we apply the optical flow method to roughly localize the target object in the frame and zoom into the target area to improve the accuracy of keypoint detection. Specifically, we first use the L-K optical flow algorithm to find displacements of the pixels between the target frame and the initial frame [19], and then employ a fixed size 512×512 pixel anchor to crop the key image area where the object is located. The cropped image will be fed into the neural network for further 2D pose estimation.

2) 2D Keypoint Detection: We apply a convolutional-based encoder-decoder for keypoint detection. Due to the huge search space and the difficulty of convergence, the input size of the model is set to be $160 \times 160 \times 3$. Consequently, the image frames captured by the camera must be resized, so that they can be fed into the model. The output of the model is the predicted 160×160 heatmap of each keypoint. Specifically, we use ResNet50 as the backbone of our visual feature encoder, this model is provided by tensorflow and is pre-trained on ImageNet [20]. We freeze the first two residual blocks at the head of ResNet and only update the parameters of the other layers during the training process. Meanwhile, the decoding part is a 3-layer upsampling structure, each containing a deconvolution layer, an activation layer, and a batch normalization layer.

3) Data Labeling: Since there is no public dataset for 6D object pose estimation using RFID-CV fusion methodology, we collect data ourselves using different moving objects. The details of the dataset will be described in Section IV. To efficiently label the object, we label a few reference points (u_x, u_y) (*e.g.*, 3 or 4) of the object on every frame, which can be combined together with the raw image to form a heatmap. Each pixel location (x_i, y_i) in the heatmap stores a value representing the probability of that location being a keypoint, which can be calculated with $\exp\left(-\frac{(x_i-\mu_x)^2+(y_i-\mu_y)^2}{\delta^2}\right)$, where δ is the artificially set hyperparameter of the network. To alleviate labeling noise, we use the average coordinates of two points with largest heat value as the position of keypoints.

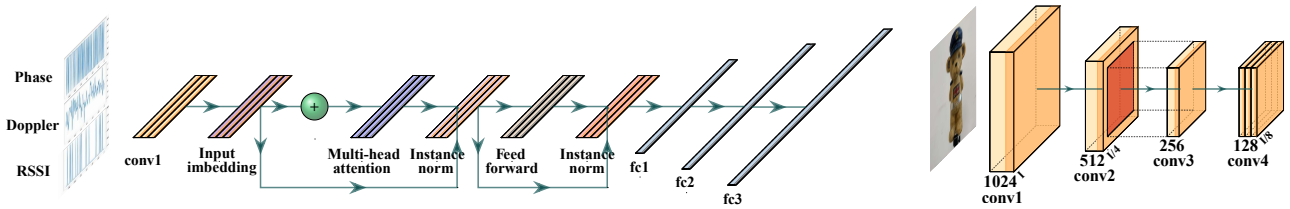
C. 3D Keypoint Detection

Given the localization of 2D keypoints, the remaining challenge is to expand 2D keypoints to 3D through predicting depth values. Inspired by the success of data-driven 3D reconstructing architectures, we design a multi-modal deep learning methodology to deeply integrate the RFID and Image data to generate a depth map of the scene. The supervised monocular depth estimation is actually an ill-conditioned problem. That is, a small pixel shift in 2D space might be corresponding to a large change in the pose of the object in 3D space. For this reason, most monocular depth estimation algorithms use a heavy backbone to extract features from the image. However, these solutions rely on extremely strict requirements for data quality, and their robustness is relatively poor. To address the above issues, this paper proposes a depth estimation network based on RFID-CV fusion. The proposed model can fuse RFID data features and RGB features to expand the 3D depth information of the RGB image. As illustrated in Fig. 3, the proposed multi-modal encoder-decoder structure can be divided into three parts: the RFID encoder, the image encoder, and the multi-modal decoder. The RFID encoder is used to extract RFID low-level signal features; the image encoder is used to extract image features; and the multi-modal decoder is used to fuse RFID data features and image features so that we can generate an improved depth maps.

1) RFID Encoder: Although low-level RFID signals like phase can be used to locate tags, these indicators suffer from hardware characteristics, multipath and noise issues, making it difficult to directly use the data for quantitative analysis. To fill this gap, we utilize multiple channels, which correspond to low-level signal characteristics, namely phase offsets, the RSSI value, and the doppler value received by the reader. We design a novel depth encoder to obtain highly robust RFID features for depth map reconstruction. It enables the network to incorporate the most relevant depth information among the three signal sequences with transformer structures.

Fig. 4(a) shows the detailed structure of the RFID feature encoder. Specifically, the input size of the RFID encoder is 3×132, indicating the signal sequences recorded by the reader, namely phase offsets, the RSSI value and the Doppler value. The input signals are firstly sent to a convolutional layer containing three 1×1 convolution kernels. Next, the results of the convolution are sent to a multi-head attention mechanism as the input of query, key, and value, which are equal to the shape of the input. After being multiplied by itself, they are sent to the softmax layer to calculate their covariance matrix with itself and multiply the covariance matrix with its own data to get the weight map. We link the weight map and output through the residual for obtaining a feature map with weights. The weight is expressed as the degree of similarity between the data. We can enhance or weaken the value of each predicted data according to the similarity between each predicted value and other data in the sequence. The similarity pixels are used in both training and prediction, while the dissimilar pixels are ignored. After that, the feature map is expanded into a one-dimensional vector and sent to the three fully connected layers containing 128, 1280 and 12800 neurons. Finally, a vector with a length of 12800 is turned into a 10×10 time-length feature map with a channel number of 128.

2) Image Encoder: The structure of the image encoder is shown in Fig. 4(b). We use convolutional neural networks to extract the spatial features of the $160 \times 160 \times 3$ RGB image. The proposed structure leverages downsampling (*i.e.*,



(a) RFID encoder (b) RGB encoder

Fig. 4. Illustrating model architectures used for RFID and image feature extraction

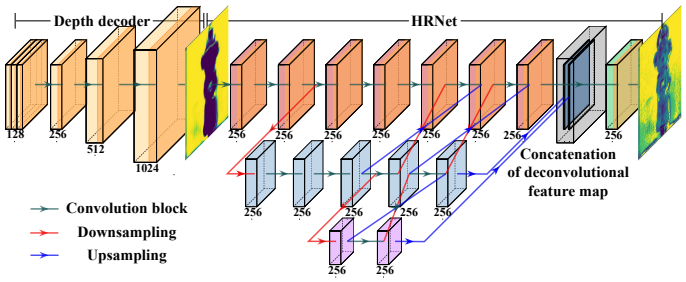


Fig. 5. Illustrating the structure of depth decoder and HRNet

polling operation), which is part of the classic segmentation network Unet, to gradually reduce the image size. This process helps the filters in the deeper layers to focus on a larger receptive field. Each downsampling module contains 64 3×3 effective convolution kernels. After passing the input RGB image through 3 stacked downsampling layers, the size of feature map is 20×20 with a channel number of 64. To improve the accuracy of depth estimation, we hope the feature maps obtained by the image encoder and the depth encoder are closely correlated. However, since we do not have any restrictions on the position of the attached RFID tag, it is likely that the tag is not in the field of view of the camera. As a result, the correlation cannot be directly measured. This problem leads to the unavailability of the perceptual region extraction algorithm such as RPN and other target detection methods. To this end, we use a sliding window to extract a set of 10×10 candidate perceptual regions from all of the 20×20 feature map. The selected candidate perceptual regions are sent to a convolutional layer containing 128 1×1 convolution kernels, in which the group convolution algorithm is adopted to merge the candidate perceptual regions to obtain the 10×10 ROI (Region of Interest) areas. The ROI areas are fed into feature maps, and then further pass through the sigmoid activation function and batch normalization layer to obtain a normalized 10×10 ROI area with 128 channels.

3) Multi-modal Fusion for Depth Estimation: In recent years, multi-modal data fusion has attracted widespread attention. Many prior algorithms directly concatenate data from different sensors, but they ignore the inherent correlation among different sources of data. To this end, we propose a novel RFID-CV fusion network based on Kalman filtering. As shown in Fig. 5, the red lines and the blue lines respectively represent downsampling and upsampling, while the remaining lines represent the convolution process. Unlike the posterior algorithms that fuse the RFID and depth maps to estimate scene information, we fuse the features of RFID and RGB data to directly regress the depth information of the original RGB image. The model inputs are two $10 \times 10 \times 128$ feature maps obtained from the depth encoder and image encoder, respectively. Since the feature maps pass through the batch normalization layer, we assume that there is a random vector x, and the unbiased estimate based on the observation of the camera and RFID is x_1 and x_2, respectively. Considering that these two data features both obey a Gaussian distribution with a mean of 0 and a variance of 1, we thus have the observation equations of the two systems as follows:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + w, \\ y_1(k) = c_1(x_1) + v_1, \\ y_2(k) = c_2(x_2) + v_2, \end{cases} \tag{5}$$

where w is the system noise, v_1 and v_2 are random variables representing the measurement noise about RFID and image, respectively. Based on the above unbiased estimator, we can calculate the error covariance matrix of the two devices. The matrix weighted fusion estimation expression of x is given by: $\hat{x} = \sum A_i \hat{x}_i$ and A satisfies: $s.t. \sum A_i = I$. Assuming that the obtained covariance matrix is $covar$, we then stipulate that the evaluation index E is the trace of the error covariance matrix, *i.e.*, $E = tr(covar)$. Our goal is to minimize E subject to the aforementioned constraints.

The traditional Kalman filter uses the Lagrangian multiplier method to solve the optimization problem. However, the confidence of the filter should be encoded in a covariance noise parameter which is difficult to set manually and should dynamically adapt to the new measurement. To solve this, we propose a neural network to learn the weight matrix of the Kalman filter, and further predict the feature map in the next scale. Specifically, we first combine the RFID data and the image, and then send them into a channel fusion layer with three 1×1 kernels to integrate the ROI of RFID and image. The normalized matrix, which is the output of a convolutional layer containing a softmax activation function and a 1×1 convolution kernel, is used to calculate the Kalman gain. On the other hand, we fuse the attention features to obtain a weighted feature map, which is also sent to a convolutional layer containing 256 1×1 kernels to restore the channel information. Similar to the residual estimation in the classical Kalman filter, we introduce a residual block structure in the network to connect the input and the output. Thereby a deconvolution operation with a step size of 2 is performed,

which means the scale is doubled. Different from the classical Kalman filter, since there is no timestamp in our network, we use different scales to represent for the changing time stamps. We expand the existing feature map twice as the predicted value. Then we find the image with a same scale in the image encoder, and combine it with the RFID data which is enlarged to the same scale as the observation value. We repeat the above process to obtain another observation and prediction, until the scale of the generate feature map is same as the origin image. To update the Kalman gain and the covariance matrix, we calculate the residual of the Kalman filter. Similar to UNet, we introduce a skip connection structure fusing the features of the upsampling stage and the downsampling stage. We not only calculate residual of the Kalman filter, but also obtain the feature map in the next scale. In this way, we can obtain high-resolution features to accelerate the convergence of the network. Finally, we send the feature map to the convolutional layer containing a 1×1 convolution kernel to obtain a predicted image with the same size as the ground truth. The loss function of our network architecture consists of three following parts:

$$\begin{cases} l = l_1 + l_2 + l_3, \\ l_1 = -[y_i \log(\hat{y_i}) + (1 - y_i) \log(1 - \hat{y_i})], \\ l_2 = |y_{\text{pred}} - y|_2, \\ l_3 = \min tr(P_k), \end{cases} \tag{6}$$

where l_2 is the mean square error between the prediction result and the label, l_1 indicates the cross-entropy function between the prediction result and the label, and it is used to prevent overfitting, l_3 is the evaluation index of Kalman gain, y is the ground-truth, X is the prediction of the network and P is the Kalman gain covariance matrix. By minimizing the trace of the error covariance matrix, RFID data and RGB data are integrated and the network's attention is focused on the area where RFID data and RGB data are most correlated. In this paper, the network adopts the adam optimizer, which adapts the learning rate for each weight of the neural network. We set the batch size to 6 and train the network with 160 epochs to obtain a coarse-grained depth maps model. The depth map produced by the model has limitations such as blurred edges and large model pixel errors. To address the above issues, we remove the last 1×1 convolutional layer, and send the output feature map to a HRnet16. The high-resolution and low-resolution features of the HRnet network architecture can be fully integrated and maintained, enabling fine-grained depth maps at high resolution and clean edges. However, the complexity of the network increases as the accuracy increases, and the difficulty of training network convergence and the time complexity of the algorithm also increases on a large scale.

Ideally, after aligning the RGB image and scene depth map, we can obtain the 3D coordinates of the keypoints by adding depth information to the 2D keypoints. Unfortunately, the depth map captured by the Kinect has many missing pixels with zero values. If the 2D keypoints happen to be localized in these pixels, we will fail to obtain the depth information. To this end, we apply a threshold-based bilinear interpolation algorithm to complement the missing depth values with the neighbouring pixels with valid values. Specifically, for an arbitrary pixel in depth map, if a pixel value is less than a threshold, we use four interpolation neighboring points to compute the missing values $f(x, y)$ as follows:

$$f(x, y) \approx \sum_{i=1}^{2} \sum_{j=1}^{2} \frac{(-1)^{i+j} f(x_i, y_j)(x_j - x)(y_i - y)}{(x_2 - x_1)(y_2 - y_1)}. \tag{7}$$

Hence, we obtain the 3D coordinates of the keypoints at the missing pixels through the above-mentioned bilinear interpolation. As illustrated in Fig. 6, we compare RC6D with two classical methods including Baseline and FastDepth, the dark space of Baseline [21] and FastDepth [22] represents the failure of depth prediction. On the contrary, in RC6D, we eliminate most of the unpredictable space through Kalman filter, then further refine the depth map by interpolation. Our depth map can restore not only keypoint locations, but also depth information, which further demonstrates the possibility of using RFID data for depth estimation.

D. 6D Pose Estimation

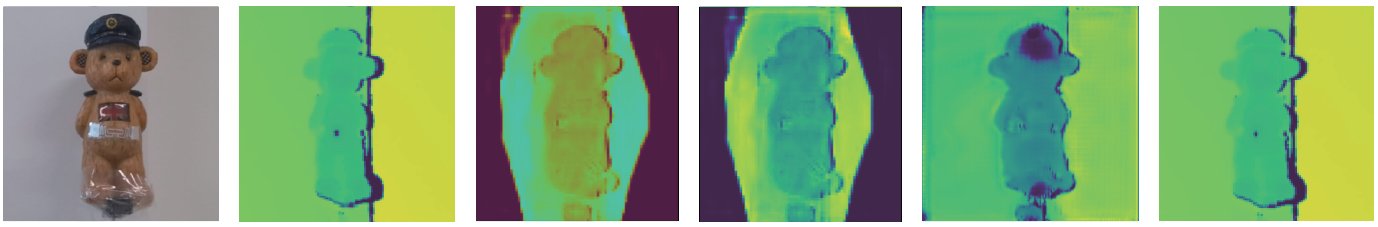
After obtaining the 2D and 3D keypoints, we match them using EPnP algorithm to estimate the 6D pose. Here we use a common calibration method [23] to calibrate the camera and obtain the camera's intrinsic matrix. Thus, we can earn the camera intrinsic according to the camera calibrate, which can be further used to resolve PnP (Perspective-n-Point) problems, which describes how to estimate the pose of the object when we know n 3D space points and their projection positions. In this paper, the number of keypoints (*i.e.*, labels) is usually more than 4, thus we apply the EPnP algorithm to fuse the 2D and 3D keypoints to obtain the 6D pose of the target [24].

IV. Evaluation

In this section, we conduct experiments to evaluate the performance of the RC6D system. We first describe the system implementation and the experiment settings. Then we conduct experiments to evaluate the performance of our system under various conditions and present the experimental results in three aspects: performance of 2D keypoint detection in RC6D, performance of 3D keypoint detection in RC6D, and performance of 6D pose estimation in RC6D. Finally, we compare RC6D with the state-of-the-art systems.

A. Implementation

The hardware components of RC6D system consists of an Impinj Speedway R420 reader, a Laird S9028PCR reader antenna, several E41C Impinj tags, an Azure Kinect, a laptop, and a high-performance server Powerleader PR2730G with an Nvidia Tesla P100 GPU. In terms of software components, we adopt the Octane C# sdk to develop a program running on the laptop to configure LLRP messages by controlling the RFID reader to report low-level signal data Phase, RSSI, Doppler with embedded spatial information. Meanwhile, the Azure Kinect is used for capturing the RGB images used for 2D keypoint detection and depth imaging for obtaining the ground-truth depth map. We leverage Azure Kinect Sensor



(a) Input image (b) Ground truth (c) Baseline (d) Fastdepth (e) Ours (no HRNet) (f) Ours

Fig. 6. Qualitative comparison of the reconstructed depth map and the ground truth value generated from Kinect

TABLE I
IMPACT OF RESNET PARAMETERS ON 2D KEYPOINT DETECTION

Backbone	Input Size	Deconv layers	Var	AP
ResNet50	80×80	3	10	85.26
ResNet50	80×80	3	20	80.40
ResNet50	80×80	3	5	83.34
ResNet50	80×80	3	1	79.74
ResNet50	160×160	**3**	**10**	**93.33**
ResNet50	120×120	3	10	89.27
ResNet50	80×80	3	10	85.26
ResNet50	40×40	3	10	69.77
ResNet50	160×160	**3**	**10**	**93.33**
ResNet50	160×160	4	10	80.14
ResNet50	160×160	2	10	77.51
ResNet50	160×160	1	10	66.28
ResNet50	160×160	**3**	**10**	**93.33**
ResNet18	160×160	3	10	91.26
VGG19	160×160	3	10	85.3
VGG16	160×160	3	10	79.28

SDK for capturing depth and RGB image, synchronization controls and some basic file operations. We conduct extensive experiments with five distinctive objects with various shape, material and colors to evaluate the performance of RC6D. For each object, we collect 300 sets of RFID and Kinect data in different positions and orientations. The object is placed about $1\,\mathrm{m}$ away from the RC6D system, and moves forward and back within the range of $\pm 0.2\,\mathrm{m}$.

B. Performance of RC6D in 2D Space

We conduct experiments to evaluate the impact of four factors on 2D keypoint detection, including backbone, input size, deconvolutional layer number and variance. In this section, we also compare the 2D keypoint detection with state-of-the-arts.

1) Impact of ResNet Parameters: In this set of experiments, we evaluate the influence of backbone, input size, deconvolutional layer number and variance of ground truth heatmap on 2D keypoint detection, respectively. Tab. I shows the impact of different parameters on 2D keypoint detection accuracy. The optimal parameters that reaches the maximum accuracy are bolded in the table.

2) Comparison with State-of-the-Arts: In this section, we compare the 2D keypoint detection with CPM and Hourglass with two metrics, including accuracy and PCK (Percentage of Correct Keypoints) [25]. PCK measures the proportion of the correct estimation of the keypoints. which calculates the percentage of detections that fall within a normalized distance of the ground truth. Fig. 7 shows that our proposed RC6D performs better than two algorithms on both standards.

C. Performance of RC6D in 3D Space

In 2D space, we evaluate the impact of four factors: backbone, input size, deconvolutional layer and variance. In 3D space, we will focus on our evaluation on the impact of other two factors: object subject and environmental condition. In this section, we use RMSE (Root Mean Squared Error) and SqRel (Square relative error) as the standards of grading to measure the ability of the depth map generation accuracy of RC6D. RMSE is a traditional metric for measuring regression errors. The squared term of SqRel penalizes larger depth errors. In this paper, we define a good Scene depth map generation is less than 6 for RMSE, and is less than 1.25 for SqRel.

1) Impact of Different Object Subjects: For each subject of the object, we collect 300 groups of data. Fig. 8 shows the result of the five objects. The scores of the bottle, milk box, box and bear are good, while the score of the disinfectant (DSF) exceeds the given range. It is because that the similarity of the color between the disinfectant and the background, which will be further discussed in the next section.

2) Impact of Environmental Conditions: To verify RC6D's immunity to different environmental conditions (*e.g.*, camera and background), we evaluate RC6D in three different conditions. In this experiment, an object (*e.g.*, the bottle) moves along the same trajectory at a same speed. For each condition, we collect 10 groups of data. Fig. 9 shows the detection error, we observe that the same trend of the score according to the standards, and the scores of both standards exceed our given range. This is reasonable since the RGB camera (*e.g.*, Kinect) cannot capture the object when the object color is similar to the background color or there is occlusion.

3) Comparison with the State-of-the-Arts: We compare the performance of RC6D in 3D keypoint detection with BaseLine and FastDepth. Since we are the first to estimate 6D pose combining RFID with CV, we design this baseline which using like Mask RCNN+UNet framework to detect object position in image and classify the object classes and generate depth map to compare with RC6D. As shown in Fig. 10, we see that the depth estimation results of other objects except disinfection water are relatively stable, which is caused by the fact that disinfection water is greatly affected by the environment. In addition, we compare the performance of RC6D with/without HRNet. The latter directly uses the rough maps of the Kalmen filter as the output of the model, thus the absence of HRNet makes the score of SqRel increase. In addition, since RC6D is a real-time 6D pose estimation system, it is important to

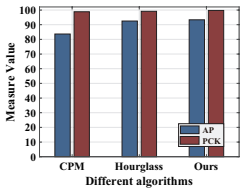


Fig. 7. 2D keypoint accuracy

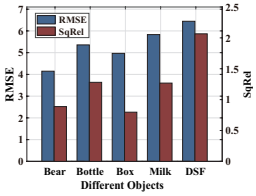


Fig. 8. RMSE v.s. objects

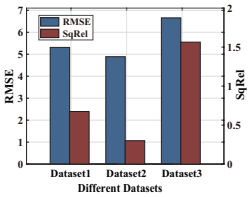


Fig. 9. RMSE v.s. data

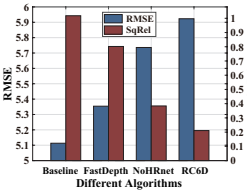


Fig. 10. RMSE v.s. alg.

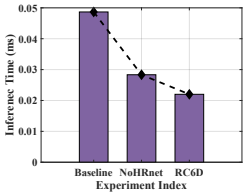
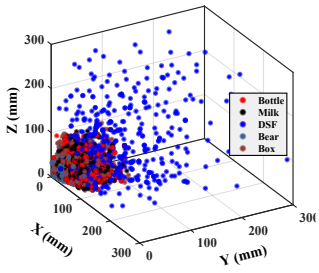
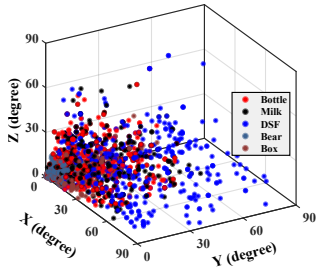


Fig. 11. Running time

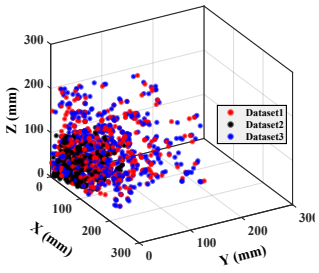


(a) Localization error

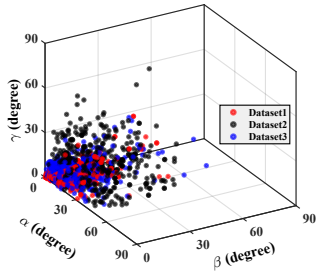


(b) Orientation error

Fig. 12. Impact of different objects



(a) Localization error



(b) Orientation error

Fig. 13. Impact of different datasets

evaluate the system's running time. We measure the running time by feeding a same video to the three methods, the results are shown in Fig. 11. We observe that RC6D is much faster than the Baseline, and it will be even faster without HRNet. This is because the addition of the HRnet fine-grained module consumes a lot of computing resources.

D. Performance of RC6D in 6D Space

We first show the matching results of 6D poses through RC6D in Fig 14. Similar to 3D keypoint detection, in this section, we conduct experiments on the two same factors: object subject and environment condition. In the 6d experiment, we use the calculated 6D pose and the real object 2D keypoint coordinates and the object 3D keypoint coordinates obtained from the depth map taken by Kinect. The absolute value of the 6 dimensions of the pose obtained by the PnP algorithm. The value error is used as the standard for evaluating positioning accuracy. As a frame of reference, it represents all instances of objects in the same category. In this section, we measure the angle and the displacement of the x-direction, the y-direction and the z-direction, and then calculate the absolute value error.

1) Impact of Object Subjects: As shown in Fig. 12, we evaluate the five types on the six scales, respectively. We can see that the displacement localization accuracy of the box and the bear is less than $10\,\mathrm{cm}$, while the angular localization accuracy is less than $15°$, and the angular localization accuracy of the disinfectant is even as high as $90°$, and the displacement localization error is as high as $20\,\mathrm{cm}$. This is due to the color of the disinfectant and the background color difference is too small, resulting in too much error in the depth map generation and 2D keypoint detection, resulting in error accumulation. As aforementioned, in our experiments, the color of the disinfectant is similar to the color of the background, thus there is limitation on the estimation of the disinfectant. This shows that the localization accuracy of this paper is heavily dependent on the accuracy of 2D keypoint detection and the accuracy of depth map generation, and it will easily lead to the phenomenon of error accumulation

2) Impact of Environmental Conditions: Again we show the impact of the various environmental conditions on 6D pose estimation. As shown in Fig. 13, the result is similar to that of the 3D keypoint detection, the accuracy is lower when the camera and background changes or there is occlusion, which is due to the change of the environment background. To be specific, it may affect the 2D keypoint detection accuracy, and makes some keypoints fail to be detected due to occlusion, which further reduces the localization accuracy. However, compared to 3D keypoint detection, there may be accumulation of errors in 6D pose estimation, which makes it more sensitive to the environment. This can be observed from Fig. 13 that when there is occlusion, the accuracy of 6D pose estimation is 0.832, which is lower than the accuracy in 3D keypoint detection.

3) Comparison with State-of-the-Art: We design this set of experiment to evaluate the importance of RFID. All of the existing 6D pose estimation approaches in category-level are dependent on 3D point cloud, which is masked since it is difficult to collect around view 3D point cloud about object. However, collecting 3D point cloud information needs multiple equipment settings, which is expensive and difficult to employ. To solve this, we propose to combine RFID and CV for 6D pose estimation. No prediction of the object mask is needed because there is only one single moving object in the view, instead, we change the branch to the encoder-decoder structure of UNet to generate object depth image. And, since the Baseline only depends on the image, it needs to identify the category of the object before estimating the 6D poses. Steps of 6D pose estimation according to the Baseline are as follows: First, it detects the object position in an image and predicts the object category. Then it detects the 2D keypoints, which is similar to RC6D. Next, when detecting the 3D keypoints, the Baseline uses the 2D keypoint coordinates as well as a combination of MASK RCNN and UNet generated map.

(a) Bottle (b) Milk box (c) Bear (d) Box (e) Disinfectant

Fig. 14. Matching of keypoints through RC6D

Finally, Baseline uses the PnP algorithm to calibrate the 6D pose. To sum up, RC6D is superior to all existing 6D pose estimation methods in all aspects of accuracy, time complexity, robustness, and generality ability.

V. Related Work

Through the thorough review of related works, we summarize and classify them into three categories below.

A. CV-based 6D Pose Estimation

CV-based methods generally extract labels from 3D models for training models and then achieve the estimation. Pavlakos *et al.* [14] predicts the keypoints by image segmentation, which strongly depends on the CAD model and needs much time on estimation. Deep-6D [6], SSD-6D [4], and Yolo-6d [5] convert the 6D pose estimation problem into a target detection problem. however, these methods pay too much attention to target detection but ignore the difference between 2D and 3D space. As aforementioned, the above instance-level methods have extremely high requirements for data, where CAD models are required for both reasoning and deployment. Consequently, it is difficult to deploy them in practice. ISA [26] adapts to the distribution offset caused by the difference in shape, and eliminates the changes in texture, lighting, pose, etc. NOCS [9] proposes a new context-aware technology to generate a large amount of full space mixed reality data to train the network, which can robustly estimate the pose and size of invisible object instances in the real environment. 6-PACK [10] is a deep learning method for category-level 6D object pose tracking on RGB-D data. Similar to the instance-level methods, the category-level methods also require the CAD model data for training, and their accuracy is relatively low.

B. Non-CV-based 6D Pose Estimation

Non-CV-based estimation methods estimate 6D poses through extra equipments (*e.g.*, IMU and motor). One mainstream method among the non-CV based methods makes use of IMU (Inertial Measurement Unit). AprilTags [13] implements fuse visual and inertial data, thus calculate the accurate 3D position of AprilTag relative to the camera. Nowka *et al.* [16] considers a commonly used gyroscope-based kinematic constraint and presents a novel accelerometer-based kinematic constraint in inertial motion tracking of kinematic chains. These methods cannot accurately estimate the 6D poses that they have a large noise. Some other methods use various equipments to achieve 6D pose estimation. 3D-OmniTrack [27], Orientation-Aware RFID Tracking [28], RED [29] tracks the position and polarization of the object through a polarization-sensitive phase model in an RFID system. However, all these non-CV based methods need extra equipment and are difficult to deploy in realistic scenarios.

C. Multi-Modal Fusion-based 6D Pose Estimation

TagVision [30] is a multi-modal fusion algorithm, which organically combines the position information given by the CV sub-system and the phase data returned by the RFID sub-system. Nguyen *et al.* [7] and Liang *et al.* [8] propose to integrate multiple sensors to estimate 6D poses, while the deployment is complex and difficult to realize. Unlike the above multi-modal fusion methods having poor ability on dealing with multi-path, our proposed RC6D is robust to different environmental conditions.

VI. Conclusion

In this paper, we proposed the RFID-CV fusion based 6D pose estimation (RC6D) system, which jointly leverages RFID and computer vision data to estimate the 6D pose of the target object. Three key technical challenges were addressed when implementing the RC6D system. First, under the challenging condition of estimating 6D poses within the CAD model, an RFID-CV fusion neural network was proposed to predict the depth of the scene. Second, to reduce manpower of labeling, we proposed another neural network to automatically label the keypoints. Third, we used RFID to directly identify objects, rather than using auxiliary AI techniques which are complex and inaccurate. Extensive experiments demonstrate that the localization error of RC6D is less than $10\,\mathrm{cm}$ with a probability higher than 90.64% and its orientation estimation error is less than 10° with a probability higher than 79.63%. The target objects can be arbitrarily chosen by the user as needed and are not limited to what we exemplified in this paper. The proposed 6D pose estimation system has the potential to be applied in various IoT scenarios.

Acknowledgment

This work was supported by NSF China (No. 62002259, 62072221, 62032017, 61772251, 42050105, 62020106005, 62061146002), 2021 Tencent AI Lab RhinoBird Focused Research Program (No: JR202132), and the Program of Shanghai Academic/Technology Research Leader under Grant No. 18XD1401800.

REFERENCES

[1] L. Manuelli, W. Gao, P. Florence, and R. Tedrake, "kpam: Keypoint affordances for category-level robotic manipulation," *arXiv preprint arXiv:1903.06684*, 2019.

[2] 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.

[3] M. K. Bekele, R. Pierdicca, E. Frontoni, E. S. Malinverni, and J. Gain, "A survey of augmented, virtual, and mixed reality for cultural heritage," *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 11, no. 2, pp. 1–36, 2018.

[4] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1521–1529.

[5] B. Tekin, S. N. Sinha, and P. Fua, "Real-time seamless single shot 6d object pose prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 292–301.

[6] T.-T. Do, M. Cai, T. Pham, and I. Reid, "Deep-6dpose: Recovering 6d object pose from a single rgb image," *arXiv preprint arXiv:1802.10367*, 2018.

[7] T. H. Nguyen, T.-M. Nguyen, and L. Xie, "Range-focused fusion of camera-imu-uwb for accurate and drift-reduced localization," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1678–1685, 2021.

[8] Y. Liang, S. Müller, D. Schwendner, D. Rolle, D. Ganesch, and I. Schaffer, "A scalable framework for robust vehicle state estimation with a fusion of a low-cost imu, the gnss, radar, a camera and lidar," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 1661–1668.

[9] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category-level 6d object pose and size estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2642–2651.

[10] C. Wang, R. Martín-Martín, D. Xu, J. Lv, C. Lu, L. Fei-Fei, S. Savarese, and Y. Zhu, "6-pack: Category-level 6d pose tracker with anchor-based keypoints," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 10 059–10 066.

[11] X. Li, H. Wang, L. Yi, L. J. Guibas, A. L. Abbott, and S. Song, "Category-level articulated object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3706–3715.

[12] J. Papon and M. Schoeler, "Semantic pose using deep networks trained on synthetic rgb-d," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 774–782.

[13] M. Li and C. Zhang, "A spatial pose measurement scheme by using imu and apriltag technologies," in *2018 IEEE International Conference on Information and Automation (ICIA)*, 2018, pp. 1048–1052.

[14] G. Pavlakos, X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis, "6-dof object pose from semantic keypoints," in *2017 IEEE international conference on robotics and automation (ICRA)*, 2017, pp. 2011–2018.

[15] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "Pvnet: Pixel-wise voting network for 6dof pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4561–4570.

[16] D. Nowka, M. Kok, and T. Seel, "On motions that allow for identification of hinge joint axes from kinematic constraints and 6d imu data," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 4325–4331.

[17] T. Boroushaki, J. Leng, I. Clester, A. Rodriguez, and F. Adib, "Robotic grasping of fully-occluded objects using rf perception," *arXiv preprint arXiv:2012.15436*, 2020.

[18] Z. Zhou, L. Shangguan, X. Zheng, L. Yang, and Y. Liu, "Design and implementation of an rfid-based customer shopping behavior mining system," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2405–2418, 2017.

[19] X. Shi, H. Cai, M. Wang, G. Wang, B. Huang, J. Xie, and C. Qian, "Tagattention: Mobile object tracing with zero appearance knowledge by vision-rfid fusion," *IEEE/ACM Transactions on Networking*, vol. 29, no. 2, pp. 890–903, 2021.

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[21] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[22] D. Wofk, F. Ma, T.-J. Yang, S. Karaman, and V. Sze, "Fastdepth: Fast monocular depth estimation on embedded systems," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6101–6108.

[23] 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.

[24] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 8, pp. 930–943, 2003.

[25] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 466–481.

[26] C. Sahin and T.-K. Kim, "Category-level 6d object pose recovery in depth images," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.

[27] C. Jiang, Y. He, S. Yang, J. Guo, and Y. Liu, "3d-omnitrack: 3d tracking with cots rfid systems," in *2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2019, pp. 25–36.

[28] C. Jiang, Y. He, X. Zheng, and Y. Liu, "Orientation-aware rfid tracking with centimeter-level accuracy," in *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2018, pp. 290–301.

[29] Y. He, Y. Zheng, M. Jin, S. Yang, X. Zheng, and Y. Liu, "Red: Rfid-based eccentricity detection for high-speed rotating machinery," *IEEE Transactions on Mobile Computing*, vol. 20, no. 4, pp. 1590–1601, 2021.

[30] C. Duan, X. Rao, L. Yang, and Y. Liu, "Fusing rfid and computer vision for fine-grained object tracking," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2017, pp. 1–9.