IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 December 2023, accepted 20 December 2023, date of publication 1 January 2024, date of current version 5 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3348781

RESEARCH ARTICLE

Occlusion-Robust Pallet Pose Estimation for Warehouse Automation

VAN-DUC VU[1], DINH-DAI HOANG[2], PHAN XUAN TAN[3], (Member, IEEE), VAN-THIEP NGUYEN[1], THU-UYEN NGUYEN[1], NGOC-ANH HOANG[1], KHANH-TOAN PHAN[1], DUC-THANH TRAN[1], DUY-QUANG VU[1], PHUC-QUAN NGO[1], QUANG-TRI DUONG[1], ANH-NHAT NGUYEN[1], AND DINH-CUONG HOANG[1]

[1]ICT Department, FPT University, Hanoi 10000, Vietnam
[2]Toyohashi University of Technology, Aichi, Toyohashi 441-8580, Japan
[3]College of Engineering, Shibaura Institute of Technology, Tokyo 135-8548, Japan

Corresponding author: Dinh-Cuong Hoang (cuonghd12@fe.edu.vn)

ABSTRACT Accurate detection and estimation of pallet poses from color and depth data (RGB-D) are integral components many in advanced warehouse intelligent systems. State-of-the art object pose estimation methods follow a two-stage process, relying on off-the-shelf segmentation or object detection in the initial stage and subsequently predicting the pose of objects using cropped images. The cropped patches may include both the target object and irrelevant information, such as background or other objects, leading to challenges in handling pallets in warehouse settings with heavy occlusions from loaded objects. In this study, we propose an innovative deep learning-based approach to address the occlusion problem in pallet pose estimation from RGB-D images. Inspired by the selective attention mechanism in human perception, our developed model learns to identify and attenuate the significance of features in occluded regions, focusing on the visible and informative areas for accurate pose estimation. Instead of directly estimating pallet poses from cropped patches as in existing methods, we introduce two feature map re-weighting modules with cross-modal attention. These modules effectively filter out features from occluded regions and background, enhancing pose estimation accuracy. Furthermore, we introduce a large-scale annotated pallet dataset specifically designed to capture occlusion scenarios in warehouse environments, facilitating comprehensive training and evaluation. Experimental results on the newly collected pallet dataset show that our proposed method increases accuracy by 13.5% compared to state-of-the-art methods.

INDEX TERMS Pose estimation, robot vision systems, intelligent systems, deep learning, supervised learning, machine vision.

I. INTRODUCTION

The next generation of industrial revolution requires a wide range of developments regarding sensing and perception technologies which would enable intelligent systems for warehousing applications [1], [2], [3]. These systems should be endowed with high flexibility and versatility to adapt to the rapidly changing market needs. This challenge can be met by further integrating intelligent vision systems. Intra-logistics applications typically comprise an array of operations, most of which deal with transport, loading, unloading, storing and moving pallets in a warehouse. As a result, pallet recognition and localization are crucial in warehouse automation [4], [5], [6].

Generally, the detection and localization of pallets is a widely researched topic in the field of automation and robotics [7], [8], [9], [10], [11], [12]. The aim of this research is to develop systems and algorithms that can accurately detect and locate pallets in real-world environments, allowing for more efficient and automated warehouse operations.

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Cano.

A presentation [7] was made regarding one of the earliest systems to use visual information for recognizing pallets and determining their posture. The algorithm incorporates a verification technique for predictions that involves projecting the geometry of the pallet model onto the image from its expected 3D location. In [8], and [9], the position and orientation of the pallet are determined through its size and edge features that were extracted from a color image captured by a camera. The authors of [10], [11] introduce a reliable technique for recognizing pallets by employing fiducial markers that are located on the pallets themselves. By utilizing a combination of range measurements from lidar sensors (Light Detection and Ranging) and color images from vision sensors, [12] demonstrate a more accurate estimation of pallet pose in unstructured environments. Despite numerous proposed techniques, the existing solutions are not considered to be robust and accurate enough for widespread adoption in the industry. The existing methods face challenges due to varying environmental conditions, such as changes in lighting, occlusions, and diverse pallet orientations, leading to difficulties in consistently achieving precise and reliable pallet detection and localization across different warehouse settings. Additionally, the reliance on single-sensor modalities or markers may limit adaptability to complex, real-world scenarios, contributing to the lack of robustness and accuracy in these approaches.

Advancements in visual recognition have led to the development of a group of data-driven approaches that use deep networks for determining the position and orientation of objects from RGB or RGB-D inputs [13], [14], [15]. These methods have been successful in recent years due to their ability to learn complex representations from image and depth data. They have demonstrated a noticeable improvement in both speed and accuracy on well-known datasets such as YCB-Video [13] or LineMOD [16], [17]. Despite their success, these methods have faced challenges when it comes to estimating pallet pose in warehouse environments. This is due to a shortage of large datasets of pallets for model training, and the complexity of the warehouse environments. The occlusion problem, in particular, poses a significant hurdle as pallets are frequently heavily obstructed by other objects. While the issue of occlusion has been extensively studied in tasks such as object detection [18], [19] and segmentation [20], [21], the domain of pose estimation for occluded pallets within warehouse environments remains a relatively unexplored area.

In this study, we present a tailored deep learning approach for achieving occlusion-robust pallet pose estimation from RGB-D images. By combining appearance information from RGB and geometry data from point clouds, our method pioneers the use of cross-modal attention maps. The application of attention mechanisms to pallet pose estimation is unprecedented, marking a distinct contribution to the field. The novelty of our approach is highlighted by the introduction of cross-modal attention mechanism specifically tailored to address the unique challenges posed by pallets. The uniform appearance of pallets can impede the identification of specific features, especially in scenarios of partial occlusion. Traditional attention maps, relying on distinctive features, face challenges in such situations. Our cross-modal attention mechanism provides a nuanced solution by effectively discerning and eliminating redundant features in occluded areas, a crucial innovation for enhancing accuracy, particularly in cluttered scenes with multiple pallets, where conventional methods may falter. Our contribution includes two attention-based filtering modules, namely geometry-aware visual feature re-weighting (GAV-FR module) and visual-aware geometric feature re-weighting (VAG-FR module). The GAV-FR module adapts the spatial attention block from [22], [23] by calculating the attention map using geometry-aware visual features, not directly applying it to the visual feature map. Similarly, the VAG-FR module, inspired by [24], [25], computes the attention map from geometry-aware visual features. These cross-modal attention modules enable our model to assign varying weights to features based on their significance to pose estimation. Additionally, we introduce a large-scale RGB-D pallet dataset, providing ground truth poses with 6 degrees of freedom (6DOF), serving as a valuable resource for advancing pallet pose estimation algorithms.

The main contributions of this work are: (1) A deep learning approach for robust 6D object pose estimation from RGB-D images, capable of handling severe occlusion and multiple pallets in cluttered scenes; (2) Two feature map re-weighting modules that effectively filter out features from occluded regions and background, enhancing the accuracy of pose estimation; (3) The introduction of a large-scale RGB-D pallet dataset, which includes ground truth poses with 6 degrees of freedom (6DOF), providing a valuable resource for training and evaluating pallet pose estimation algorithms.

The remainder of this article is organized as follows. Section II, Related Work, delves into Object Pose Estimation from RGB-D Data (II.A) and Dataset for 6D Object Pose Estimation (II.B). In Section III, Methodology, we introduce the Backbones (III.A) and discuss our innovations: Geometry-aware Visual Feature Re-weighting (III.B), Visual-aware Geometric Feature Re-weighting (III.C), and Fusion and Pose Regression (III.D). Section IV, Evaluation, comprises an overview of the Pallet RGBD Dataset (IV.A), Implementation Details (IV.B), Evaluation Metric (IV.C), and Results (IV.D), while also extending the assessment to established benchmarks in Evaluation on Common Benchmarks (IV.E). Finally, in Section V, Conclusions, we summarize key contributions, findings, and outline potential avenues for future research.

II. RELATED WORK

In this section, we review relevant works, specifically focusing on existing RGB-D based methods and datasets for 6D Object Pose Estimation.

A. OBJECT POSE ESTIMATION FROM RGB-D DATA

Classical approaches process RGB-D data as input, extracting 3D features, and subsequently conduct correspondence grouping with hypothesis verification [16], [17], [26], [27], [28], [29]. However, these features are either handcrafted or acquired through optimizing surrogate objectives such as reconstruction, rather than the ultimate goal of determining the 6D pose. Such features designed by human experts would have limited performance in changing lighting conditions and scenes with severe occlusion. Recently, with the tremendous growth in machine learning and deep learning, Deep Neural Network (DNN) based methods have been introduced into this task and have shown promising advancements. Xiang et al. [13] present PoseCNN, a Convolutional Neural Network that estimates the initial pose from RGB images and refines it using Iterative Closest Point (ICP) [30] on the point cloud. PoseCNN estimates the 3D translation of an object by identifying its center in the image and determining the 3D center's distance from the camera. The rotation of the object is estimated by regressing convolutional features to a quaternion representation. Additionally, the authors present ShapeMatch-Loss, a novel loss function to address symmetrical objects, which improves the results for such objects. However, this approach requires the use of ICP algorithm [30] for refinement, which limits its feasibility for real-time applications due to its slow processing speed. Instead, the approaches [31], [32] use information from Convolutional Neural Networks (CNNs) applied to RGB images to supplement the geometry reasoning on bird's-eye-view (BEV) images of point clouds. However, these methods fail to consider the advantages of including geometry information in learning RGB representations and ignore the pitch and roll of object pose. Furthermore, standard 2D CNNs are inadequate in handling contiguous geometry reasoning.

To fully utilize the advantages of both data sources, the works [14], [33], [34], [35], [36], [37] extract features from each separately using specialized networks and then fuse them appropriately. They use a CNN to extract features from RGB images and a point cloud network such as PointNet [38] or PointNet++ [39] to extract features from the point clouds. The resulting features are then combined to improve the accuracy of pose estimation. This approach combines the strengths of each data source to better handle occlusions and variability in the scene. The fusion of the features can be achieved through various methods. Wang et al. [14] propose DenseFusion module to fuses RGB values and point clouds at the per-pixel level. The fused features are then used to regress pose parameters. This per-pixel fusion scheme allows the model to explicitly consider local visual information and geometric details, resulting in promising performance on benchmarks such as YCB-Video and Occluded-LINEMOD datasets [13], [17]. However, DenseFusion relies on the output of a semantic segmentation network in its initial stage to identify and classify objects. Challenges arise when occlusion interferes with accurate object boundary identification, leading to potential segmentation errors and pose estimation failures. In contrast, methods like [36], [37] operate directly on input scene images without relying on segmentation results. These methods employ the DenseFusion module to fuse features and predict keypoints, subsequently using least-squares fitting to estimate object poses. While keypoint-based approaches excel on standard datasets, we empirically observed their limitations for objects like pallets, where accurate keypoints are challenging to detect, especially in occluded scenes. Our work aligns closely with DenseFusion [14]; however, instead of relying on segmentation results, we employ a detection framework to obtain object bounding boxes [40], [41]. We acknowledge that these bounding boxes often encompass the pallet and loaded objects, as well as the background. To filter out irrelevant features for pallet pose estimation, we introduce two attention-based filtering modules: the geometry-aware visual feature re-weighting (GAV-FR module) and the visual-aware geometric feature re-weighting (VAG-FR module). These proposed feature map re-weighting modules effectively enhance accuracy by eliminating features from occluded regions and background.

B. DATASET FOR 6D OBJECT POSE ESTIMATION

In recent years, the number and scale of datasets for object pose estimation have increased rapidly. LineMOD [16] is a widely-used dataset for 6D pose estimation, which includes objects located in complicated backgrounds. The dataset was captured using PrimeSense Carmine RGB-D sensor, and includes 15 objects, two of which are symmetrical. Each sequence is labeled with the poses of a single object, with the target objects having either no occlusion or very minimal occlusion in the ground truth poses. Brachmann et al. [17] expanded the LineMOD dataset to overcome the shortage of occluded test data by adding more ground truth poses for all the modeled objects in a single test sequence. This supplementary annotation introduces testing scenarios with varying degrees of occlusion that are difficult, and enables the assessment of multiple object localizations. T-LESS [42] is also a popular dataset designed to simulate a common robotic bin-picking situation. It includes 30 industry-relevant objects with no discriminative color or reflectance properties, featuring symmetrical shapes and sizes. A unique aspect of this dataset compared to others is its focus on industrial objects and some of the objects are parts of others. YCB-Video Dataset [13] is another well-known collection of data that features 21 YCB objects with different textures and shapes. The dataset includes 92 RGB-D videos of a subset of these objects, which have been labeled with 6D pose and instance semantic masks. This dataset is considered challenging due to the varying lighting conditions, image noise, and occlusions present in the captured videos. While the above datasets and benchmarks are useful for evaluating and comparing algorithms, they are not directly applicable to the unique challenges and requirements of warehouse environments. Therefore, there is a need for

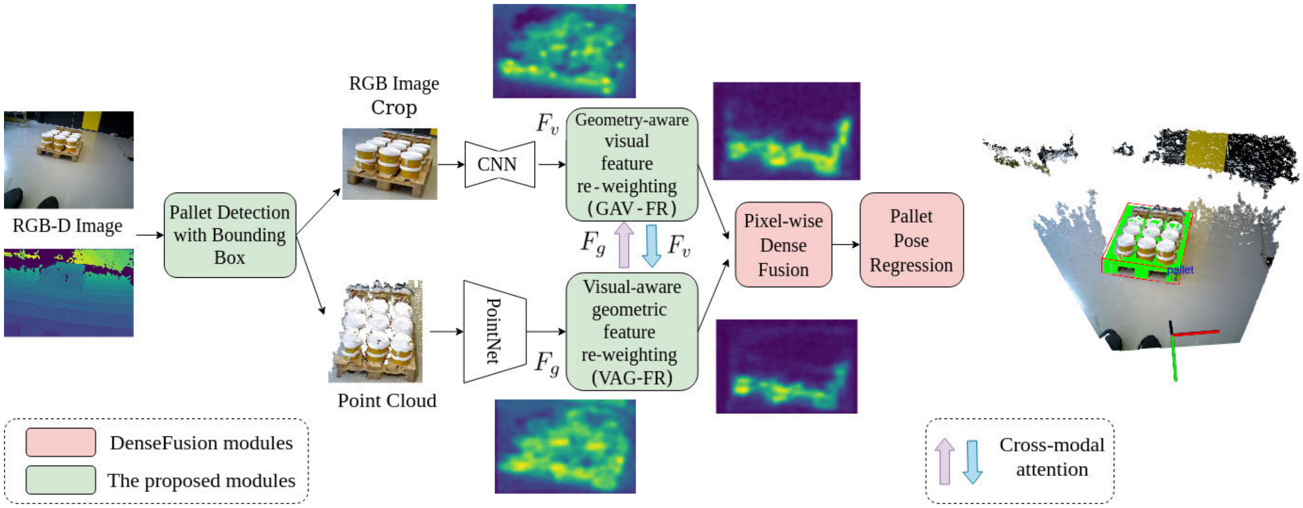
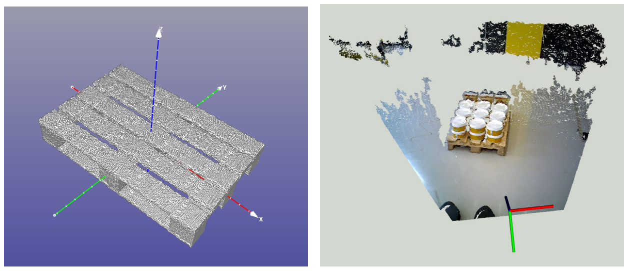


FIGURE 1. **Overview of our network architecture. Initially, bounding boxes are generated from RGB images using an off-the-shelf object detector. The RGB and point cloud data sources are then processed separately to extract features, which are subsequently passed through attention-based re-weighting feature map modules. These modules selectively emphasize relevant information, enhancing the precision of pose estimations. Next, a dense fusion network [14] is employed to combine the extracted features and generate dense feature embeddings at the pixel level. Finally, the pose regression module is utilized to achieve accurate pallet pose estimation.**



(a)

(b)

FIGURE 2. **Coordinate systems. (a) Pallet model coordinate system. (b) Camera coordinate system.**

datasets specifically designed for warehouse applications. The Amazon Robotics Challenge [43] has released several datasets that include real-world sensor data collected from Amazon warehouses, along with corresponding ground-truth labels for task manipulation. As the field of warehouse automation continues to grow, it is likely that we will see more efforts to develop and release datasets specifically designed for this application domain. These datasets will be critical for enabling researchers and practitioners to develop and evaluate effective machine learning algorithms and automation systems that can improve warehouse operations and worker safety. In this paper, we contribute a large RGB-D pallet dataset with 6DOF ground truth poses. The dataset is intended to support the development of solutions for accurately estimating pallet poses in confined warehouse environments that are encountered in picking tasks. The dataset's comprehensive scope and design provide researchers with the versatility to use the data in developing deep learning models for automating warehouse operations.

III. METHODOLOGY

Given an RGB-D image, we aim to detect pallets and determine their orientations and translations in a three-dimensional (3D) space. The problem we address shares similarities with the task of 3D pallet detection, where the objective is to determine the oriented 3D bounding boxes of pallets based on data captured by sensors. These bounding boxes encompass the entire pallet and are defined by parameters such as size (height, width, length), center position, and orientation. However, for warehouse applications such as pallet picking, the information provided by bounding boxes is often insufficient. Instead of detecting pallets, it would be more beneficial to align a complete 3D model of the pallet with the partially observed data. To achieve this, we assume the availability of an accurate 3D model of the pallet, and we establish a coordinate system $\mathcal{O}$ in the 3D space of the model (Figure. 2). The pose of the pallet is represented by a rigid transformation from the pallet coordinate system to a reference coordinate system $\mathcal{G}$, typically the camera coordinate system. This rigid transformation is comprised of a rotation matrix $R \in SO(3)$ and a translation vector $t \in \mathbb{R}^3$, $\xi = [R|t]$. We propose a pallet pose estimation network as shown in Figure 1.

Our approach tackles the task by separately processing the two data sources and leveraging a dense fusion network [14] to extract dense feature embeddings at the pixel level. Our key innovation lies in the feature filtering stage with cross-modal attention. Here, we propose two attention-based filtering modules to effectively filter out irrelevant features, including geometry-aware visual feature re-weighting (GAV-FR module) and visual-aware geometric feature re-weighting (VAG-FR module). The GAV-FR module leverages the spatial attention block from [22], [23]. However, distinguishing itself from the approach in [22], and [23], we calculate the attention map using geometry-aware visual features instead of directly applying it to the visual feature map. Similarly, for the VAG-FR module, inspired by the self-attention block in [24], [25], our departure involves computing the attention map using geometry-aware visual features rather than directly applying it to the visual feature map. By incorporating cross-modal attention modules, we enable the model to assign varying weights to different features based on their significance to the pose estimation task. This mechanism facilitates the model in focusing on the most informative and relevant visual and geometric cues while mitigating the impact of noise or irrelevant information. This can potentially lead to more accurate and reliable pose estimation results, especially in challenging scenarios with occlusions, cluttered backgrounds, or varying lighting conditions.

A. BACKBONES

Given an RGBD image, the first stage takes the color image as input and performs a pallet detection task. We employ YOLOv8 [44] to generate bounding boxes, which are then used to crop depth and color images. In the next step, the cropped RGB image is fed into convolutional neural networks to extract visual features. We adopt ResNet-50 [45] as the backbone network, generating the visual feature map $F_v \in \mathbb{R}^{C\times H\times W}$, where C is the number of channels, and H and W are the height and width of the map, respectively. To extract geometric features F_g from the cropped depth image, we convert the depth pixels into a 3D point cloud with known camera intrinsics. This point cloud serves as input to a backbone network based on PointNet [38]. The backbone network enriches each 3D point with high-dimensional features, denoted as $\mathcal{P} = \{pi\}_{i=1}^N$, $F_g = \{f_i\}_{i=1}^N$, where $p_i = [x_i; f_i]$. Here, $x_i \in \mathbb{R}^3$ represents the point's location in 3D space, and $f_i \in \mathbb{R}^K$ (K channels) is a feature vector associated with the point.

B. GEOMETRY-AWARE VISUAL FEATURE RE-WEIGHTING

Given the feature map $F \in \mathbb{R}^{C\times H\times W}$ extracted from the cropped RGB image as input, this module employs spatial attention mechanisms [22], [23] to re-weight features within the map. Spatial attention is designed to assign importance weights to different spatial locations in a feature map. These weights reflect the significance of each location in contributing to the final prediction. However, in contrast to [22], [23], we compute the attention map using geometry-aware visual features F_{gv} instead of directly applying it to the visual feature map (Figure 3). This approach allows us to effectively leverage the complementary information from both RGB and depth modalities. Given the well-aligned RGBD image, we utilize the 3D point clouds as a conduit to link visual and geometric features. For each pixel in the feature map with its xyz coordinate, we identify its m nearest points from the point cloud and collect the corresponding geometric features. Subsequently, we resize the geometric features to match the channel size of the visual feature and employ max pooling to integrate them. The resultant integrated point feature is then concatenated with the corresponding visual feature and processed through a shared MLP, generating the geometry-aware visual features F_{gv}. In the next step, we aggregate information from the feature map F_{gv} through average-pooling and max-pooling operations along the channel axis (global pooling). This process generates two new feature maps: $F_{gv}^{avg} \in \mathbb{R}^{1\times H\times W}$ representing the average-pooled features, and $F_{gv}^{max} \in \mathbb{R}^{1\times H\times W}$ representing the max-pooled features. Subsequently, we concatenate F_{avg} and F_{max} and apply a standard convolution layer, resulting in a spatial attention map M. The spatial attention M is computed as following:

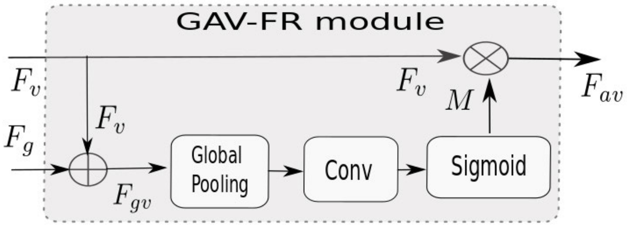


FIGURE 3. The architecture of the GAV-FR module. ⊗ denotes element-wise multiplicatio. ⊕ denotes the feature integration (more detail in section III-B).

$$M = \sigma(f^{k\times k}([AvgPool(F_{gv}); MaxPool(F_{gv})])) \quad (1)$$

where σ denotes the sigmoid function and $f^{k\times k}$ represents a convolution operation with the filter size of $k \times k$. We empirically chose $k = 7$ following the setting in [22]. The attention map M assigns scores to each feature according to its spatially varying importance, which is caused by redundant information in the feature map (such as background or other objects). Using the attention map M, the visual feature map F_v is re-weighted as following:

$$F_{av} = F_v \odot M \quad (2)$$

where $\odot$ is the Hadamard product.

C. VISUAL-AWARE GEOMETRIC FEATURE RE-WEIGHTING

Point cloud data, an unordered set of 3D points, poses challenges for traditional convolutional neural networks (CNNs) designed for grid-like structures. Self-attention models [25], [46] provide a promising solution, excelling in handling complex and irregular structures without relying on point connections. Inspired by their success, we introduce a module for point cloud attention learning incorporating spatial attention. Utilizing the self-attention mechanism, we reveal geometric correlations between 3D points, extracting features relevant to pallet pose estimation. By explicitly considering spatial relationships, like distances or angles, we focus on crucial geometric features for accurate pallet pose estimation.

As detailed in Section III-A, for extracting geometric features from the cropped depth image, we initially convert depth pixels into a 3D point cloud using known camera intrinsics. This point cloud serves as input to the PointNet backbone network [38]. The backbone network enriches each 3D point with high-dimensional features, denoted as $\mathcal{P} = \{p_i\}_{i=1}^N$, $F_g = \{f_i\}_{i=1}^N$, where $p_i = [x_i; f_i]$. Here, $x_i \in \mathbb{R}^3$ represents the point's location in 3D space, and $f_i \in \mathbb{R}^F$ is a feature vector associated with the point. To enhance F_g, we incorporate visual cues F_v into geometric features, resulting in visual-aware geometric features F_{vg}. For each geometric feature with its 3D point coordinate, we identify its l nearest points and collect their corresponding visual features. Subsequently, we utilize max pooling to integrate these neighboring visual features and employ shared Multi-Layer Perceptrons (MLPs) to resize them to the same channel size as the geometric feature map. The enriched points $\{p_i\}_{i=1}^N$ with enhanced features are then fed into our self-attention module to re-weight the features. Following [24], [25] the self-attention module is defined as follows:

$$y_i = \sum_{p_j \in \mathcal{P}(i)} (\alpha(\gamma(p_i, p_j) + \delta) \odot \beta(p_j)) \tag{3}$$

The subset $\mathcal{P}(i) \subseteq \mathcal{P}$ encapsulates a localized cluster of points surrounding a central point p_i within the broader point cloud $\mathcal{P}$. Within this context, the functions α, γ, δ, and β play distinct roles. α, the mapping function, calculates attention weights that determine the importance of relationships between points in the local neighborhood $\mathcal{P}(i)$. The relation function γ quantifies the interaction between two points, p_i and p_j, employing subtraction to produce a single vector representing the features of both points. Meanwhile, the position encoding function δ introduces spatial context by computing parameterized encodings based on the spatial coordinates of points p_i and p_j, allowing the model to consider their positional relationships. Finally, the pointwise feature transformation function β updates the features of neighboring points within $\mathcal{P}(i)$ based on the attention weights generated by α and the relations captured by γ. The relation function γ computes a single vector that represents the features of p_i and p_j using subtraction:

$$\gamma(p_i, p_j) = \varphi(p_i) - \psi(p_j) \tag{4}$$

Where φ and ψ are trainable transformations performed by multilayer perceptrons (MLPs). The mapping function α is an MLP consisting of two linear layers and one ReLU nonlinearity. This architecture allows the module to generate attention weights that vary spatially and across channels while maintaining computational efficiency by reducing dimensionality. To enable adaptation to local structures within the data, we use a trainable and parameterized position encoding δ, defined as:

$$\delta = \phi(x_i - x_j) \tag{5}$$

Here x_i and x_j are the 3D point coordinates for points i and j. The encoding function ϕ is an MLP with two linear layers and one ReLU nonlinearity.

D. FUSION AND POSE REGRESSION

So far, we have obtained dense features from both the image and the 3D point cloud inputs. To fuse this information, we follow the approach described in [14]. First, we associate the geometric feature of each point with its corresponding image feature pixel by projecting it onto the image plane using the known camera intrinsic parameters. This association allows us to create pairs of features, which are then concatenated and inputted into another network. This network uses a symmetric reduction function to generate a fixed-size global feature vector. Rather than relying solely on a single global feature for the estimation, we enrich each dense pixel-feature with the global densely-fused feature to provide a global context. This ensures that each per-pixel feature benefits from the overall global information. Subsequently, we feed each of these resulting per-pixel features into a final network, which predicts the 6D pose of the pallet. By combining the dense features from the image and the 3D point cloud and leveraging the geometric association between them, we effectively fuse the information to capture both local and global contexts. This approach enhances the accuracy of the pallet's 6D pose estimation.

We define the pose estimation loss as the distance between the points sampled from the ground truth pose of the object model and the corresponding points on the same model transformed by the predicted pose. Specifically, the loss to be minimized is defined as:

$$L_i^p = \frac{1}{M} \sum_{x_j \in M} \min_{0<k<M} \left| (Rx_j + t) - (\hat{R}_i x_j + \hat{t}_i) \right| \tag{6}$$

Here, x_j represents the j^{th} point among the randomly selected M 3D points from the pallet's 3D model. The ground truth pose is denoted as $p = [R|t]$, while the predicted pose, generated from the fused embedding of the i^{th} point, is denoted as $\hat{p}_i = [\hat{R}_i|\hat{t}_i]$. The loss function calculates the minimum distance between the transformed ground truth point and the predicted point, averaging over all the randomly selected 3D points. To optimize the pose estimation for all the predicted poses, we minimize the sum of losses: $L = \frac{1}{N} \sum_i L_i^p$, where N represents the total number of points from the cropped depth image.

IV. EVALUATION

This section presents the evaluation of our proposed system, conducted through experiments on a newly collected pallet dataset, which, to the best of our knowledge, is the first of its kind in this important application. Our aim is to evaluate the effectiveness of our proposed approach in utilizing available data to predict the pallet's pose. Specifically, we are interested in comparing the performance of the learned model to the baseline DenseFusion and state-of-the-art methods. Additionally, we assess the robustness of our approach to clutter and investigate the extent to which filtering modules can mitigate the negative impact of occlusions. We also compare our results to those of the most closely related works.

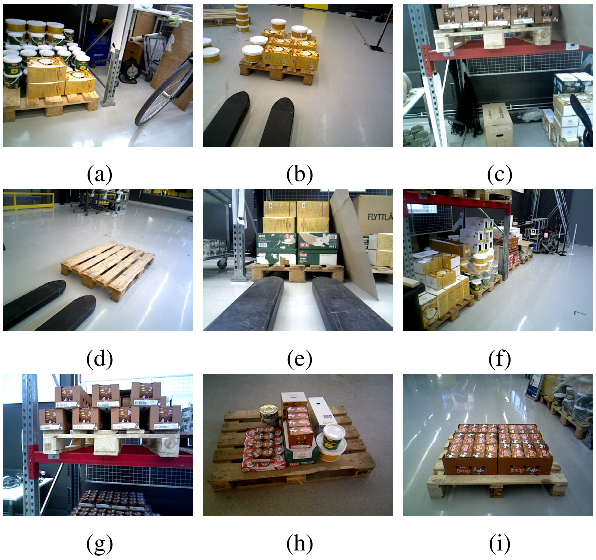

(a) (b) (c)
(d) (e) (f)
(g) (h) (i)

FIGURE 4. **Examples from the pallet dataset. Images were captured using RealSense and ASUS Xtion PRO Live cameras from various viewpoints.**

A. THE PALLET RGBD DATASET

The proposed dataset consists of 80,000 annotated RGBD images of pallets captured using RealSense and ASUS Xtion PRO Live cameras from various viewpoints. Ground truth 6D poses and instance segmentation masks are generated using the LabelFusion framework [47]. The LabelFusion framework streamlines the annotation process by employing advanced algorithms and tools that assist in the accurate generation of ground truth poses and instance segmentation masks. By leveraging the capabilities of LabelFusion, the dataset ensures a high level of precision and consistency in the annotations, reducing human error and increasing the reliability of the generated ground truth information. This dataset is specifically designed for pallet picking tasks and provides complete 6DOF ground truth poses for all images. While some existing datasets offer ground truth poses for objects in cluttered spaces, this new dataset goes further by providing object poses both with and without clutter, thereby controlling for clutter. During the data collection process, several essential controls were implemented to ensure the dataset's quality and comprehensiveness. These controls involved capturing multiple viewpoints of the pallets from various angles and additional frames to account for sensor noise. By including these additional frames, the dataset becomes more robust and better suited for training and testing perception algorithms in challenging environments. Unlike alternative datasets that reconstruct scenes, this dataset includes transformation matrices between the camera location, enabling users to reconstruct the scene according to their methods. Camera trajectories were captured using a motion capture system developed by Qualisys.[1] Calibration was performed for both the RGB-D sensor and the motion capture system. The motion capture system was calibrated using the Qualisys Track Manager (QTM) software. For RGB-D camera calibration, intrinsic parameters were estimated using a black-white chessboard pattern and the OpenCV library. Extrinsic calibration involved placing markers on the checkerboard corners and attaching spherical markers to the sensor. This enabled the estimation of the transformation between the motion capture system's pose and the RGB-D camera's optical frame. These calibration procedures ensured accurate camera trajectories and alignment with the ground truth data for further analysis. Lastly, this new dataset is tailored to the warehouse perception task and focuses on pallets. We split the dataset into 65,000 images for training and 15,000 images for testing. Figure 4 shows examples of scenes in the dataset. To the best of our knowledge, this is the first attempt to generate a real-world dataset for this crucial application.

B. IMPLEMENTATION DETAILS

In our implementation, we utilize a ResNet18 encoder followed by 4 up-sampling layers as the decoder for cropped RGB images. The output appearance feature from this encoder-decoder architecture consists of 128 channels. For point cloud feature extraction, the PointNet architecture is an MLP followed by an average-pooling reduction function, which also produces a 128-channel output. The implementations are realized by PyTorch and Python platforms on a single Nvidia GeForce RTX 2080 Ti 11GB GPU using CUDA and Linux operating system. Our pallet pose estimation framework is trained from scratch in an end-to-end manner using an Adam optimizer [52]. We train the entire network with the batch size 8 and learning rate 0.001 for 200 epochs. Figure 5 shows learning curves.

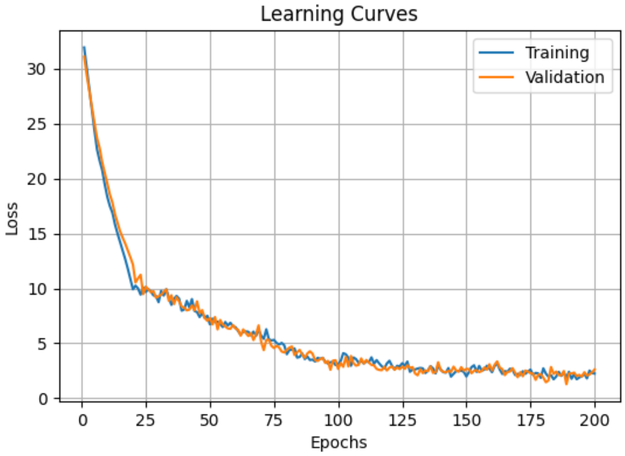
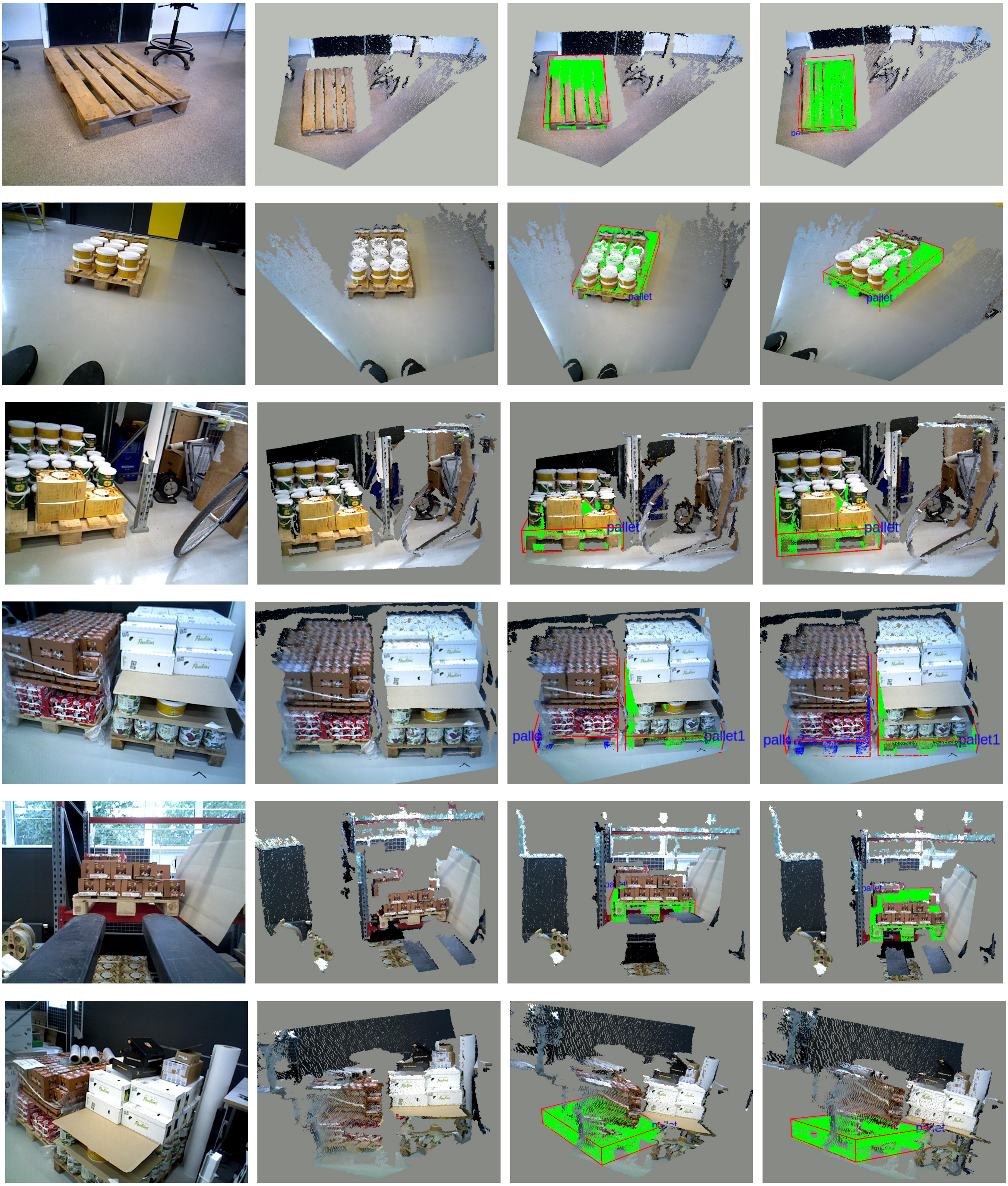


FIGURE 5. **Learning curves of our model trainning on the pallet dataset.**

C. EVALUATION METRIC

To evaluate the accuracy of an estimated pose $\hat{P}$, the widely used pose-error function Average Distance of Model Points (ADD) is used [13], [15], [48]. This involves calculating

[1] https://www.qualisys.com



(a) RGB (b) Point cloud (c) Predicted pose (d) After refinement

FIGURE 6. **Qualitative results of the proposed method. The first two columns show color images and point clouds. The third column presents the visualization of the predicted pallet pose using 3D bounding boxes, with green points representing the 3D pallet model points aligned to the scene based on the predicted pose. The fourth column showcases the visualization of the refined predicted pose using the Iterative Closest Point (ICP) algorithm [30].**

TABLE 1. Quantitative evaluation of pallet pose estimation on the newly collected pallet dataset. We report the accuracy of predictions in average precision (AP) with the ADD metric following [48]. The table compares our proposed method with various methods, including conventional methods [16], [29], [49], [50], and deep learning-based methods [13], [14], [36], [37], [51], in estimating the pose of unloaded and loaded pallets. An unloaded pallet is defined as a pallet that does not have any objects or goods loaded onto it. A loaded pallet refers to a pallet with loaded objects. Results are reported for two scenarios: without refinement (w/o) and with refinement (w/) using the Iterative Closest Point (ICP) algorithm [30]. The Runtime column indicates the execution time in milliseconds (ms) without refinement.

		Input		Unloaded Pallet		Loaded Pallet		Runtime
Method	Deep Learning	RGB	Depth	w/o refinement	w/ refinement	w/o refinement	w/ refinement	(ms)
Bohacs et al. [49] (Gray)		√		0.29	0.41	0.31	0.30	300
Bohacs et al. [49] (RGB)		√		0.33	0.43	0.30	0.38	300
Shao et al. [50]			√	0.65	0.66	0.23	0.31	2200
Hoang et al. [29]			√	0.58	0.64	0.13	0.15	3100
Hinterstoisser et al. [16]			√	0.64	0.65	0.41	0.42	1600
RePose [51]	√		√	0.62	0.66	0.33	0.36	48
PVN3D [36]	√	√	√	0.67	0.71	0.44	0.53	228
PoseCNN [13]	√	√		0.63	0.70	0.21	0.25	96
DenseFusion [14]	√	√	√	0.70	0.77	0.40	0.45	175
FFB6D [37]	√	√	√	0.66	0.72	0.37	0.41	179
Ours (without GAV-FR)	√	√	√	0.75	0.82	0.57	0.64	80
Ours (without VAG-FR)	√	√	√	0.73	0.79	0.55	0.60	79
Ours	√	√	√	**0.77**	**0.83**	**0.66**	**0.74**	87

TABLE 2. Quantitative evaluation of pallet pose estimation on the newly collected pallet dataset. We report the Area Under Curve (AUC) of the ADD metric by varying the distance threshold, with a maximum threshold of 10 cm [13], [14]. The table compares our proposed method with various methods, including conventional methods [16], [29], [49], [50], and deep learning-based methods [13], [14], [36], [37], [51], in estimating the pose of unloaded and loaded pallets. An unloaded pallet is defined as a pallet that does not have any objects or goods loaded onto it. A loaded pallet refers to a pallet with loaded objects. Results are reported for two scenarios: without refinement (w/o) and with refinement (w/) using the Iterative Closest Point (ICP) algorithm [30]. The Runtime column indicates the execution time in milliseconds (ms) without refinement.

		Input		Unloaded Pallet		Loaded Pallet		Runtime
Method	Deep Learning	RGB	Depth	w/o refinement	w/ refinement	w/o refinement	w/ refinement	(ms)
Bohacs et al. [49] (Gray)		√		0.40	0.51	0.40	0.48	300
Bohacs et al. [49] (RGB)		√		0.43	0.52	0.41	0.50	300
Shao et al. [50]			√	0.73	0.76	0.35	0.41	2200
Hoang et al. [29]			√	0.70	0.74	0.21	0.27	3100
Hinterstoisser et al. [16]			√	0.75	0.77	0.51	0.53	1600
RePose [51]	√		√	0.73	0.78	0.45	0.49	48
PVN3D [36]	√	√	√	0.80	0.83	0.56	0.64	228
PoseCNN [13]	√	√		0.73	0.79	0.31	0.37	96
DenseFusion [14]	√	√	√	0.78	0.85	0.50	0.56	175
FFB6D [37]	√	√	√	0.75	0.83	0.47	0.53	179
Ours (without GAV-FR)	√	√	√	0.84	0.90	0.65	0.71	80
Ours (without VAG-FR)	√	√	√	0.82	0.87	0.64	0.69	79
Ours	√	√	√	**0.86**	**0.92**	**0.75**	**0.83**	87

the average distance between vertices of the pallet model in the ground-truth pose and vertices of the model in the estimated pose, using the closest point distance method described in previous studies [53]. If this average distance is less than 2cm, the 6D pose estimate is considered to be a true positive. This metric is important because it measures the accuracy of predictions under the minimum tolerance for robot picking, which is typically 2cm for most forklifts. Additionally, we compute the Area Under Curve (AUC) of the ADD metric by varying the distance threshold, with a maximum threshold of 10 cm [13], [14], [36].

D. RESULT

Our experiments examine the performance of our proposed approach and other methods on the newly collected pallet dataset. We compares methods based on two different input modalities. Some methods use only depth information [16], [29], [50], [51], some use only RGB information [13], [49], while some use both depth and RGB information [14], [36]. We chose to compare our approach with these methods as they represent the state-of-the-art in object pose estimation, particularly demonstrating noteworthy results on standard datasets such as YCB-Video or Occluded-LINEMOD. Additionally, the publicly available implementations of these methods ensure a fair and reproducible benchmark.

The experiments are based on publicly available implementations by the authors [13], [14], [36], [51], and on our own implementation of the rest. Figure 6 and and Table 1 and 2 report results. The table presents the quantitative evaluation results for pallet pose estimation on the newly collected pallet dataset. The evaluation is conducted on both unloaded pallets (pallets without any objects) and loaded pallets (pallets with loaded objects). The table also includes information about the input types (appearance (RGB) and geometric (depth)), the use of ICP (Iterative Closest Point) algorithm [30], and the runtime in milliseconds (ms) for each method. The experiments were conducted on a single Intel Xeon E-2716G CPU running at 3.7 GHz, along with an

Nvidia GeForce RTX 2080 Ti GPU with 11GB of memory. Based on the results, the proposed method outperforms all evaluated methods in terms of accuracy, while maintaining a competitive runtime. In particular, it achieves the highest AUC 0.77 for unloaded pallets and 0.74 for loaded pallets without the use of the ICP algorithm. When using the ICP algorithm, it still achieves the highest AUC 0.83 for unloaded pallets and 0.74 for loaded pallets. It increases the AUC by 7% and 6% for unloaded pallets without and with ICP, respectively, and by 22% and 21% for loaded pallets without and with ICP, respectively. On average, it elevates accuracy by 13.5%. Comparing the runtime, the proposed method is relatively fast, with an execution time of 87 ms. These results highlight the potential of the proposed method for practical applications requiring accurate pallet pose estimation, such as warehouse automation and robotic pallet manipulation.

The approach incorporates two important modules: GAV-FR module, which focuses on re-weighting color image features, and VAG-FR module, which emphasizes 3D point cloud features. By re-weighting the color image features, GAV-FR module effectively improves the accuracy of the proposed method. This enhancement is evident when comparing the overall performance of the proposed method without GAV-FR to the performance of the complete proposed method. The inclusion of module leads to a notable increase in accuracy, particularly for loaded pallets. Specifically, the AUC improves by 9% for loaded pallets without ICP and 10% for loaded pallets with ICP, demonstrating the efficacy of the re-weighting strategy. Similarly, VAG-FR module, known as 3D Point Cloud Feature Re-weighting, contributes significantly to the accuracy improvement of the proposed method. By emphasizing informative regions within the 3D point cloud data, the module enables the method to capture and utilize geometric characteristics more effectively. Comparing the performance of the proposed method without VAG-FR to the overall performance, it is evident that the inclusion of the module leads to a substantial increase in accuracy, particularly for loaded pallets.

The proposed approach incorporates two important modules: GAV-FR module, which focuses on re-weighting color image features, and VAG-FR module, which emphasizes 3D point cloud features. By re-weighting the color image features, GAV-FR module effectively improves the accuracy of the proposed method. This enhancement is evident when comparing the performance of the proposed method with GAV-FR and without VAG-FR to other methods. The inclusion of module leads to a notable increase in accuracy, particularly for loaded pallets. Specifically, the AUC improves by 11% for loaded pallets without ICP and, demonstrating the efficacy of the re-weighting strategy. Similarly, VAG-FR module, known as 3D Point Cloud Feature Re-weighting, contributes significantly to the accuracy improvement of the proposed method. By emphasizing informative regions within the 3D point cloud data, the module enables the method to capture and utilize geometric characteristics more effectively. Comparing the performance of the proposed method without VAG-FR to the overall performance, it is evident that the inclusion of the module leads to a substantial increase in accuracy, particularly for loaded pallets.

TABLE 3. Object pose estimation results (AUC) with different attention modules, evaluated on the pallet RGBD dataset. Results are reported for two scenarios: without refinement (w/o) and with refinement (w/) using the Iterative Closest Point (ICP) algorithm [30].

	Unloaded Pallet		Loaded Pallet	
Method	w/o icp	w/ icp	w/o icp	w/ icp
Non-local [54] (2018)	0.66	0.70	0.38	0.45
Criss-cross [55] (2019)	0.70	0.76	0.42	0.49
Dual-attn [56] (2019)	0.70	0.77	0.43	0.48
Point attention [57] (2020)	0.76	0.80	0.51	0.56
W-AdaIN [58] (2022)	0.80	0.85	0.55	0.60
EMAF [59] (2023)	0.83	0.88	0.61	0.67
Ours	**0.86**	**0.92**	**0.75**	**0.83**

To further investigate the effect of the proposed cross-modal attention module, we conducted a comparative analysis with other attention modules that have demonstrated success in various tasks. These attention modules, including Non-local [54] (2018), Criss-cross [55] (2019), Dual-attn [56] (2019), Point attention [57] (2020), W-AdaIN [58] (2022), and EMAF [59] (2023), were integrated into our framework. For a fair comparison, we removed our cross-modal attention modules and trained the network with the same settings. The results are presented in Table 3. It is evident that our proposed approach consistently outperforms the alternative attention modules across both unloaded and loaded pallet scenarios. This suggests that the cross-modal attention module in our framework effectively captures and integrates information from RGB and depth modalities, leading to enhanced pose estimation performance. Even without the Iterative Closest Point (ICP) refinement, our approach demonstrates competitive performance, showcasing its inherent robustness.

Although our method excels in detecting and accurately estimating the pose of pallets, even in the presence of heavy occlusions, it's important to acknowledge certain limitations and potential challenges. The bottom row of Figure 6 illustrates scenarios where our method may encounter failures, particularly when the input measurement data is of poor quality. One significant limitation arises when the captured point clouds contain inaccurate geometric information, leading to potential prediction errors. In cases where cameras are mounted on robots for real-time applications, the challenge becomes dynamic as the robot navigates its environment. Our method, while effective, may face difficulties in situations where the robot's movement introduces variations in the observed scene, leading to suboptimal data quality. Our future work involves integrating next best view planning methods into our system. These methods involve dynamically planning the robot's movements to optimize the viewpoints for data acquisition, ensuring a continuous stream of high-quality input data. By incorporating such strategies, our method can adapt to changing environmental conditions and maintain reliable performance in dynamic scenarios.

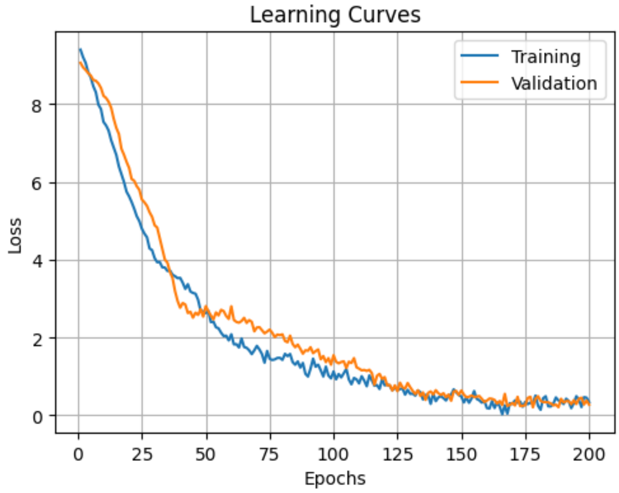


FIGURE 7. Learning curves of our model trainning on Occluded-LINEMOD dataset.

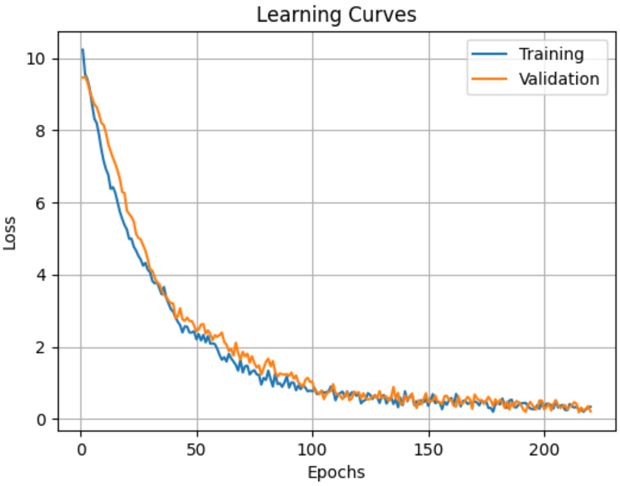


FIGURE 8. Learning curves of our model trainning on YCB dataset.

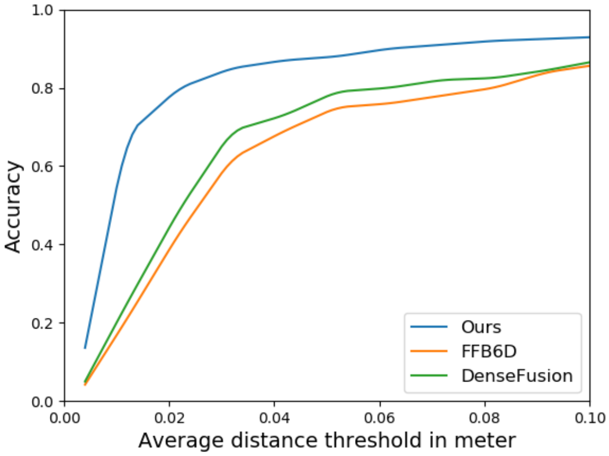


FIGURE 9. Accuracy-threshold curves with the average distance error on Occluded-LINEMOD dataset [17].

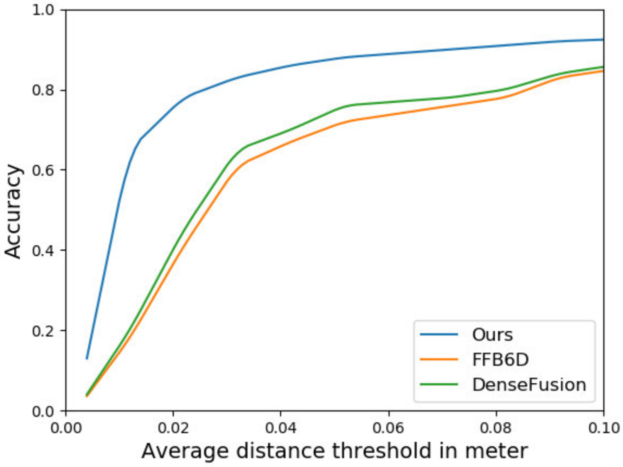


FIGURE 10. Accuracy-threshold curves with the average distance error on YCB-Video dataset [13].

E. EVALUATION ON COMMON BENCHMARKS

We conducted further experiments to evaluate the effectiveness of our proposed method on two widely recognized benchmark datasets: Occluded-LINEMOD [17] and YCB-Video [13]. The Occluded-LINEMOD dataset [17] is derived from the earlier LINEMOD dataset introduced by Hinterstoisser et al. [16]. Occluded-LINEMOD introduces additional challenges compared to LINEMOD, such as cluttered backgrounds, textureless objects, changing lighting conditions, and severe occlusions between multiple object instances. These occlusion scenarios add complexity to the task of accurate pose estimation. However, it's important to note that Occluded-LINEMOD comprises only 1214 testing images and does not provide explicit training data. Therefore, we adopted the approach used in prior works [13], which generate training data with 80,000 synthetic images. Both training and testing images have a resolution of 640 × 480 pixels. We trained our network from scratch in an end-to-end manner using the Adam optimizer [52]. Our training setup included a batch size of 8 and the incorporation of common data augmentation techniques. The initial learning rate was set to 0.001, and we trained the network for a total of 200 epochs. Learning rate decay occurred at epochs 100, 140, and 160, with corresponding decay rates of 0.1 for each step. The entire training process, until convergence, required approximately 12 hours. Figure 7 shows learning curves.

The YCB-Video dataset [13] serves as a comprehensive benchmark for 6D object pose estimation. It encompasses 21 objects with varying sizes and textures, presenting a diverse set of challenges for pose estimation algorithms. The dataset comprises approximately 130,000 real images captured from 92 video sequences and an additional 80,000 synthetically rendered images that focus exclusively on foreground objects. Precise pose annotations are provided for all objects, along with corresponding segmentation masks,

TABLE 4. Quantitative evaluation of object pose estimation on Occluded-LINEMOD dataset. We report the Area Under Curve (AUC) of the ADD metric by varying the distance threshold, with a maximum threshold of 10 cm [13], [14]. The table compares our proposed method with various methods, including conventional methods [16], [29], [49], [50], and deep learning-based methods [13], [14], [36], [37], [51], in estimating the pose of unloaded and loaded pallets. Results are reported for two scenarios: without refinement (w/o) and with refinement (w/) using the Iterative Closest Point (ICP) algorithm [30].

	Input		Occlusion (<50%)		Occlusion (>50%)		Occlusion (>70%)	
Method	RGB	Depth	w/o refinement	w/ refinement	w/o refinement	w/ refinement	w/o refinement	w/ refinement
Bohacs et al. [49] (Gray)	√		0.72	0.81	0.41	0.46	0.15	0.23
Bohacs et al. [49] (RGB)	√		0.70	0.77	0.41	0.52	0.16	0.21
Shao et al. [50]		√	0.66	0.77	0.35	0.44	0.14	0.22
Hoang et al. [29]		√	0.61	0.68	0.25	0.32	0.05	0.10
Hinterstoisser et al. [16]		√	0.80	0.90	0.54	0.56	0.21	0.28
RePose [51]		√	0.84	0.91	0.46	0.54	0.17	0.20
PVN3D [36]	√	√	0.83	0.90	0.54	0.65	0.23	0.30
PoseCNN [13]	√		0.80	0.86	0.35	0.42	0.08	0.16
DenseFusion [14]	√	√	0.83	0.90	0.51	0.60	0.20	0.25
FFB6D [37]	√	√	0.83	0.88	0.46	0.51	0.23	0.27
Ours (without GAV-FR)	√	√	0.85	0.93	0.68	0.74	0.38	0.45
Ours (without VAG-FR)	√	√	0.85	0.91	0.66	0.73	0.37	0.44
Ours	√	√	**0.86**	**0.94**	**0.77**	**0.84**	**0.41**	**0.48**

TABLE 5. Quantitative evaluation of object pose estimation on YCB-Video dataset. We report the Area Under Curve (AUC) of the ADD metric by varying the distance threshold, with a maximum threshold of 10 cm [13], [14]. The table compares our proposed method with various methods, including conventional methods [16], [29], [49], [50], and deep learning-based methods [13], [14], [36], [37], [51], in estimating the pose of unloaded and loaded pallets. Results are reported for two scenarios: without refinement (w/o) and with refinement (w/) using the Iterative Closest Point (ICP) algorithm [30].

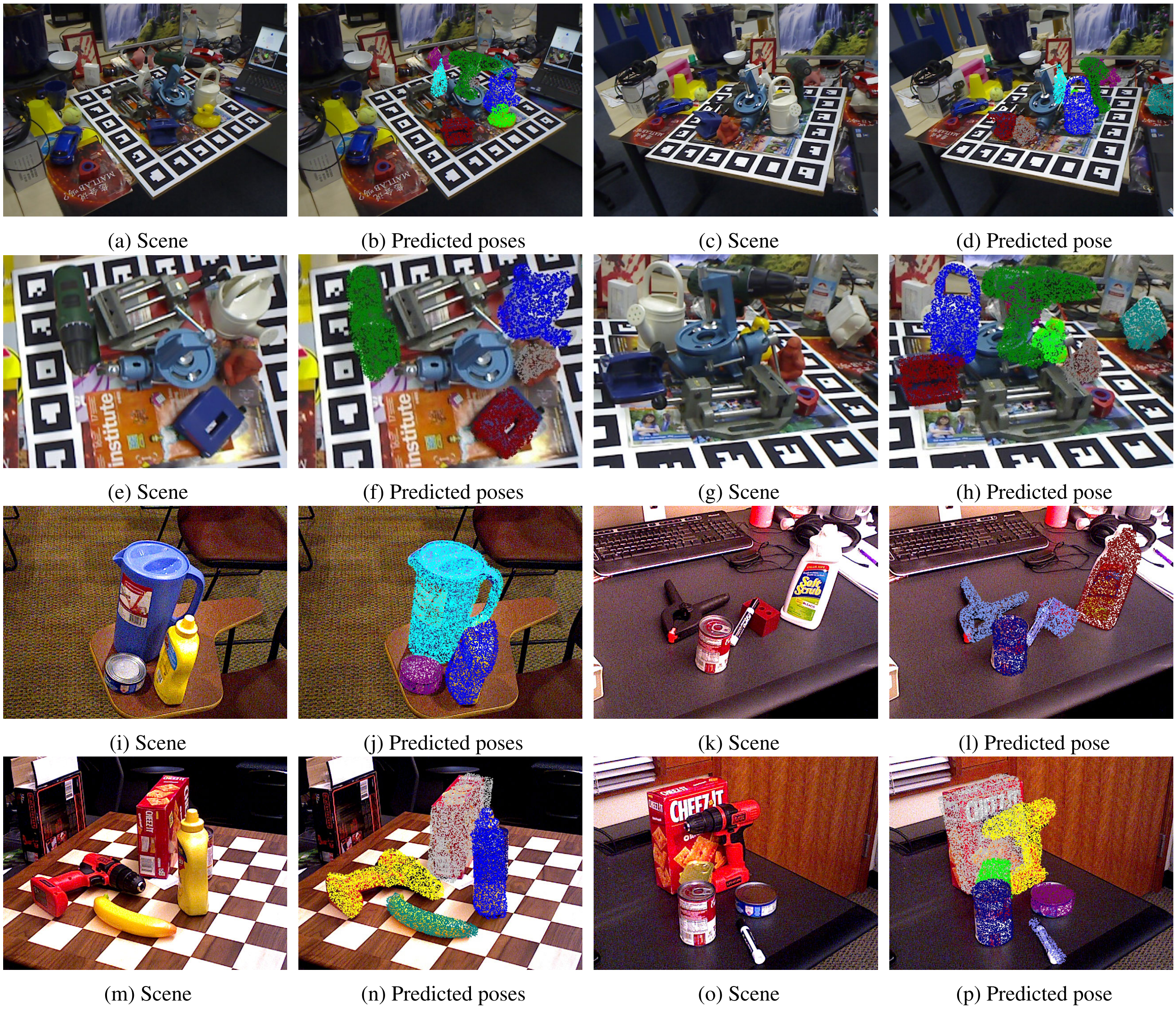
	Input		Occlusion (<50%)		Occlusion (>50%)		Occlusion (>70%)	
Method	RGB	Depth	w/o refinement	w/ refinement	w/o refinement	w/ refinement	w/o refinement	w/ refinement
Bohacs et al. [49] (Gray)	√		0.74	0.82	0.44	0.49	0.16	0.25
Bohacs et al. [49] (RGB)	√		0.72	0.80	0.43	0.56	0.18	0.22
Shao et al. [50]		√	0.69	0.78	0.37	0.46	0.15	0.23
Hoang et al. [29]		√	0.62	0.70	0.25	0.33	0.05	0.11
Hinterstoisser et al. [16]		√	0.81	0.90	0.55	0.58	0.21	0.29
RePose [51]		√	0.85	0.91	0.48	0.55	0.17	0.22
PVN3D [36]	√	√	0.87	0.92	0.58	0.69	0.24	0.30
PoseCNN [13]	√		0.81	0.88	0.35	0.43	0.08	0.17
DenseFusion [14]	√	√	0.84	0.92	0.54	0.61	0.20	0.26
FFB6D [37]	√	√	0.85	0.91	0.47	0.53	0.23	0.28
Ours (without GAV-FR)	√	√	0.87	0.94	0.70	0.75	0.40	0.46
Ours (without VAG-FR)	√	√	0.86	0.93	0.68	0.74	0.38	0.45
Ours	√	√	**0.88**	**0.96**	**0.78**	**0.85**	**0.42**	**0.49**

facilitating precise evaluation of pose estimation methods. Test images in the YCB-Video dataset exhibit a wide range of challenging factors, including diverse illumination conditions, noise, and occlusions, which significantly increase the difficulty of the pose estimation task. Following the methodology in [13], we divided the dataset into 80 videos for training, while the remaining 12 videos contributed 2,949 keyframes for testing. Similar to the Occluded-LINEMOD dataset, we employed a resolution of 640 × 480 pixels for both training and testing input images. For optimization, we used the Adam optimizer with an initial learning rate of 0.01. To aid learning, we scheduled the decay of the learning rate at epochs 120, 160, and 180, with respective decay rates of 0.1. Our entire network was trained with a batch size of 8, and common data augmentation techniques were applied. The training process spanned 220 epochs and took approximately 20 hours to reach convergence. Figure 8 shows learning curves.

We report quantitative results of object pose estimation under different levels of occlusion in Table 4 and 5. Here the levels of occlusion is estimated by calculating the invisible surface percentage of each object in the image frame following [14]. Qualitative results on the Occluded-LINEMOD and YCB-Video test sets with occlusion (>50%) are illustrated in Figures 9 and 10. These figures depict the area under the accuracy-threshold curve using the ADD metric, varying the threshold for the average distance to subsequently compute pose accuracy, with a maximum threshold set at 10cm. Our method demonstrates state-of-the-art performance, surpassing FFB6D [37] and DenseFusion [14], which are specifically designed for 6D object pose estimation using RGBD images. These experimental results indicate that our proposed approach is not only suitable for pallets but can also be applied to other objects with varying shapes and sizes.

F. RUNTIME

Table 6 provides an overview of the inference runtime measured in frames-per-second (FPS) for various pose estimation methods on a Nvidia GeForce RTX 2080 Ti GPU. The experiments utilized images of size 640 × 480 during inference, and the reported FPS values represent averages over the respective test sets. Each row corresponds to a different deep learning-based pose estimation


(a) Scene (b) Predicted poses (c) Scene (d) Predicted pose

(e) Scene (f) Predicted poses (g) Scene (h) Predicted pose

(i) Scene (j) Predicted poses (k) Scene (l) Predicted pose

(m) Scene (n) Predicted poses (o) Scene (p) Predicted pose

FIGURE 11. **Qualitative results on Occluded-LINEMOD dataset (a-h) and YCB-Video dataset (i-p).**

TABLE 6. **Inference runtime measured in frames-per-second (FPS) on a Nvidia GeForce RTX 2080 Ti GPU. Images of size 640 × 480 were used during inference where the inference FPS was averaged over the test sets.**

Method	Pallet	YCB-Video	Occluded-LMO
RePose [51]	21	10	10
PVN3D [36]	4	2	2
PoseCNN [13]	10	8	8
DenseFusion [14]	6	4	4
FFB6D [37]	6	4	4
Ours (without GAV-FR)	13	9	9
Ours (without VAG-FR)	13	9	9
Ours	12	8	8

method, including RePose [51], PVN3D [36], PoseCNN [13], DenseFusion [14], FFB6D [37], and our proposed method with and without Geometry-Aware Visual Feature Re-weighting (GAV-FR) and Visual-Aware Geometric Feature Re-weighting (VAG-FR). The runtime performance is crucial in assessing the real-time capabilities of these methods, and the table offers insights into their efficiency under the specified hardware conditions. Notably, our proposed method, incorporating both GAV-FR and VAG-FR, achieves competitive FPS values, demonstrating its potential for real-world pose estimation applications.

V. CONCLUSION

We present a robust deep learning approach for accurately estimating pallet pose from RGBD images, even in the presence of heavy occlusions. By incorporating attention mechanisms into our network architecture, we effectively filter out features from occluded regions and background, leading to enhanced prediction accuracy. To tackle the challenges posed by occlusions, we introduce two attention-based re-weighting feature map modules for both the color image and point cloud inputs. These modules selectively emphasize relevant information, resulting in more precise pose estimations. Moreover, our contribution includes the release of a large RGB-D pallet dataset with 6DOF ground truth poses. This dataset serves as a valuable resource for researchers involved in the development of solutions for accurately estimating pallet poses in confined warehouse environments, specifically in picking tasks. Through rigorous experimentation, we demonstrate that our method achieves state-of-the-art performance on the newly collected dataset, which consists of challenging occlusions. The demonstrated robustness and accuracy of our approach underscore its potential to significantly improve pallet pose estimation in real-world scenarios. The experimental results also suggest that our proposed method is not limited to pallets but can be employed effectively for objects of diverse shapes and sizes as well.

REFERENCES

[1] H. Kalkha, A. Khiat, A. Bahnasse, and H. Ouajji, "The rising trends of smart E-commerce logistics," *IEEE Access*, vol. 11, pp. 33839–33857, 2023.

[2] D.-C. Hoang, T. Stoyanov, and A. J. Lilienthal, "Object-RPE: Dense 3D reconstruction and pose estimation with convolutional neural networks for warehouse robots," in *Proc. Eur. Conf. Mobile Robots (ECMR)*, Sep. 2019, pp. 1–6.

[3] D.-C. Hoang, A. J. Lilienthal, and T. Stoyanov, "Object-RPE: Dense 3D reconstruction and pose estimation with convolutional neural networks," *Robot. Auton. Syst.*, vol. 133, Nov. 2020, Art. no. 103632.

[4] M. Hussain and R. Hill, "Custom lightweight convolutional neural network architecture for automated detection of damaged pallet racking in warehousing & distribution centers," *IEEE Access*, vol. 11, pp. 58879–58889, 2023.

[5] P. Balatti, F. Fusaro, N. Villa, E. Lamon, and A. Ajoudani, "A collaborative robotic approach to autonomous pallet Jack transportation and positioning," *IEEE Access*, vol. 8, pp. 142191–142204, 2020.

[6] A. Palleschi, M. Gugliotta, C. Gabellieri, D.-C. Hoang, T. Stoyanov, M. Garabini, and L. Pallottino, "Fully autonomous picking with a dual-arm platform for intralogistics," in *Proc. I-RIM Conf. I-RIM*, 2020, pp. 109–111.

[7] G. Garibotto, S. Masciangelo, M. Ilic, and P. Bassino, "Service robotics in logistic automation: ROBOLIFT: Vision based autonomous navigation of a conventional fork-lift for pallet handling," in *Proc. 8th Int. Conf. Adv. Robot. (ICAR)*, 1997, pp. 781–786.

[8] J. Pages, X. Armangué, J. Salvi, J. Freixenet, and J. Martí, "A computer vision system for autonomous forklift vehicles in industrial environments," in *Proc. 9th Medit. Conf. Control Autom. (MEDS)*, 2001, pp. 1–6.

[9] S. Byun and M. Kim, "Real-time positioning and orienting of pallets based on monocular vision," in *Proc. 20th IEEE Int. Conf. Tools With Artif. Intell.*, vol. 2, Nov. 2008, pp. 505–508.

[10] M. Seelinger and J.-D. Yoder, "Automatic pallet engagment by a vision guided forklift," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2005, pp. 4068–4073.

[11] M. M. Aref, R. Ghabcheloo, and J. Mattila, "A macro-micro controller for pallet picking by an articulated-frame-steering hydraulic mobile machine," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 6816–6822.

[12] N. Bellomo, E. Marcuzzi, L. Baglivo, M. Pertile, E. Bertolazzi, and M. De Cecco, "Pallet pose estimation with LiDAR and vision for autonomous forklifts," *IFAC Proc. Volumes*, vol. 42, no. 4, pp. 612–617, 2009.

[13] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes," 2017, *arXiv:1711.00199*.

[14] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "DenseFusion: 6D object pose estimation by iterative dense fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3338–3347.

[15] D.-C. Hoang, J. A. Stork, and T. Stoyanov, "Voting and attention-based pose relation learning for object pose estimation from 3D point clouds," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 8980–8987, Oct. 2022.

[16] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes," in *Computer Vision—ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5–9, 2012, Revised Selected Papers, Part I 11*. Berlin, Germany: Springer, 2013, pp. 548–562.

[17] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part II 13*. Springer, 2014, pp. 536–551.

[18] S. Gilroy, E. Jones, and M. Glavin, "Overcoming occlusion in the automotive environment—A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 23–35, Jan. 2021.

[19] A. Wang, Y. Sun, A. Kortylewski, and A. Yuille, "Robust object detection under occlusion with context-aware CompositionalNets," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12642–12651.

[20] L. Ke, Y.-W. Tai, and C.-K. Tang, "Deep occlusion-aware instance segmentation with overlapping BiLayers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4018–4027.

[21] X. Yuan, A. Kortylewski, Y. Sun, and A. Yuille, "Robust instance segmentation through reasoning about multi-object occlusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11136–11145.

[22] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[23] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6298–6306.

[24] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16239–16248.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6000–6010.

[26] R. Rios-Cabrera and T. Tuytelaars, "Discriminatively trained templates for 3D object detection: A real time scalable approach," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2048–2055.

[27] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 858–865.

[28] W. Kehl, F. Milletari, F. Tombari, S. Ilic, and N. Navab, "Deep learning of local RGB-D patches for 3D object detection and 6D pose estimation," in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III 14*. Springer, 2016, pp. 205–220.

[29] D.-C. Hoang, L.-C. Chen, and T.-H. Nguyen, "Sub-OBB based object recognition and localization algorithm using range images," *Meas. Sci. Technol.*, vol. 28, no. 2, Feb. 2017, Art. no. 025401.

[30] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," *Proc. SPIE*, vol. 1611, pp. 586–606, Apr. 1992.

[31] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D proposal generation and object detection from view aggregation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–8.

[32] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3D object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 641–656.

[33] D. Xu, D. Anguelov, and A. Jain, "PointFusion: Deep sensor fusion for 3D bounding box estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 244–253.

[34] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D object detection from RGB-D data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 918–927.

[35] K. Wada, E. Sucar, S. James, D. Lenton, and A. J. Davison, "MoreFusion: Multi-object reasoning for 6D pose estimation from volumetric fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14528–14537.

[36] Y. He, W. Sun, H. Huang, J. Liu, H. Fan, and J. Sun, "PVN3D: A deep point-wise 3D keypoints voting network for 6DoF pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11629–11638.

[37] Y. He, H. Huang, H. Fan, Q. Chen, and J. Sun, "FFB6D: A full flow bidirectional fusion network for 6D pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3002–3012.

[38] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.

[39] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5105–5114.

[40] J. Wang, X. Lin, and H. Yu, "POAT-Net: Parallel offset-attention assisted transformer for 3D object detection for autonomous driving," *IEEE Access*, vol. 9, pp. 151110–151117, 2021.

[41] S. Chen, P. Sun, Y. Song, and P. Luo, "DiffusionDet: Diffusion model for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2023, pp. 19830–19843.

[42] T. Hodan, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 880–888.

[43] C. Eppner, S. Höfer, R. Jonschkowski, R. Martín-Martín, A. Sieverling, V. Wall, and O. Brock, "Lessons from the Amazon picking challenge: Four aspects of building robotic systems," in *Proc. Robot., Sci. Syst.*, 2016, pp. 4831–4835.

[44] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[46] H. Zhao, J. Jia, and V. Koltun, "Exploring self-attention for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10073–10082.

[47] P. Marion, P. R. Florence, L. Manuelli, and R. Tedrake, "Label fusion: A pipeline for generating ground truth labels for real RGBD data of cluttered scenes," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 3235–3242.

[48] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes," in *Proc. Asian Conf. Comput. Vis.* Springer, 2012, pp. 548–562.

[49] G. Bohacs, Z. Rozsa, and B. Bertalan, "Mono camera based pallet detection and pose estimation for automated guided vehicles," in *Proc. 10th Int. Conf. Logistics, Inform. Service Sci. (LISS)*. Singapore: Springer, 2021, pp. 1–11.

[50] Y. Shao, Z. Fan, B. Zhu, J. Lu, and Y. Lang, "A point cloud data-driven pallet pose estimation method using an active binocular vision sensor," *Sensors*, vol. 23, no. 3, p. 1217, Jan. 2023.

[51] S. Iwase, X. Liu, R. Khirodkar, R. Yokota, and K. M. Kitani, "RePOSE: Fast 6D object pose refinement via deep texture rendering," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3283–3292.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015.

[53] R. Bregier, F. Devernay, L. Leyrit, and J. L. Crowley, "Symmetry aware evaluation of 3D object detection and pose estimation in scenes of many parts in bulk," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2209–2218.

[54] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[55] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.

[56] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3141–3149.

[57] M. Feng, L. Zhang, X. Lin, S. Z. Gilani, and A. Mian, "Point attention network for semantic segmentation of 3D point clouds," *Pattern Recognit.*, vol. 107, Nov. 2020, Art. no. 107446.

[58] H. Wang, M. Wang, Z. Che, Z. Xu, X. Qiao, M. Qi, F. Feng, and J. Tang, "RGB-depth fusion GAN for indoor depth completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 6199–6208.

[59] J. Yang, S. Gao, Z. Li, F. Zheng, and A. Leonardis, "Resource-efficient RGBD aerial tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 13374–13383.

VAN-DUC VU is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. His research interest includes computer vision.

DINH-DAI HOANG is currently pursuing the bachelor's degree in computer science and engineering with the Toyohashi University of Technology, Japan. His research interests include computer vision, image processing, and deep learning.

PHAN XUAN TAN (Member, IEEE) received the B.E. degree in electrical-electronic engineering from Military Technical Academy, Vietnam, the M.E. degree in computer and communication engineering from the Hanoi University of Science and Technology, Vietnam, and the Ph.D. degree in functional control systems from the Shibaura Institute of Technology, Japan. He is currently an Associate Professor with the Shibaura Institute of Technology. His current research interests include deep learning for visual computing, image and video processing, computational light field, 3D view synthesis, the multimedia quality of experience, and multimedia networking.

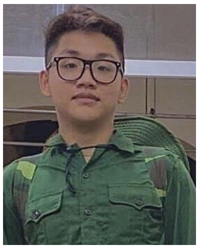
VAN-THIEP NGUYEN is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. His current focus revolves around the manipulation of objects by robots and object recognition.

THU-UYEN NGUYEN is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. Her research interest includes computer vision.

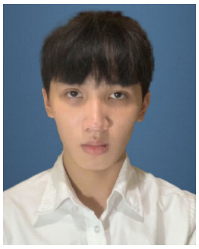
NGOC-ANH HOANG is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. His research interest includes computer vision.

KHANH-TOAN PHAN is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. His research interest includes computer vision.

DUC-THANH TRAN is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. His research interest includes computer vision.



DUY-QUANG VU is currently pursuing the B.S. degree in artificial intelligence with FPT University, Hanoi, Vietnam. His research interest includes computer vision.



PHUC-QUAN NGO is currently pursuing the B.S. degree in computing with FPT University, Greenwich Vietnam, Hanoi, Vietnam. His research interests include computer vision and the Internet of Things (IoT).

QUANG-TRI DUONG is currently pursuing the B.S. degree in computing with FPT University, Greenwich Vietnam, Hanoi, Vietnam. His research interests include computer vision and the Internet of Things (IoT).

ANH-NHAT NGUYEN received the B.S. degree in computer science from Duy Tan University, Da Nang, Vietnam, in 2012, and the M.S. degree in computer science from the Huazhong University of Science and Technology (HUST), China, in 2018. He is currently a Lecturer with FPT University, Vietnam. His research interests include image processing, information security, physical layer secrecy, radio-frequency energy harvesting, and wireless sensor networks.

DINH-CUONG HOANG received the Ph.D. degree in computer science from Orebro University, Sweden, in 2021. He is currently a Lecturer with FPT University, Greenwich Vietnam. His research interests include the intersection of computer vision, robotics, and machine learning. He is particularly interested in topics involving autonomy for robots, with a focus on perception algorithms.