Multimodal 6D Detection of Industrial Pallets, in Real and Virtual Environments, with Applications in Industrial AMRs

José Lourenço, Gonçalo Arsénio, Luís Garrote and Urbano J. Nunes
University of Coimbra, Institute of Systems and Robotics, Department of Electrical and Computer Engineering, Portugal
{garrote, urbano}@isr.uc.pt

Keywords: 6D Pallet Detection, Multimodal Deep Learning, RGB-D Based and Point Cloud Based.

Abstract: In this work we propose a multimodal approach for detecting and estimating the 6D pose of pallets, to be applied in industrial environments. The method is designed for future integration with Autonomous Mobile Robots (AMRs) for enhanced warehouse automation. Using the DenseFusion framework as basis, the proposed approach fuses RGB and Depth data using multi-head self-attention mechanisms to improve its robustness. To test the proposed methods, three datasets were developed: two virtual and one real-world indoor dataset, with varying degrees of occlusion and alignment challenges. Experimental results demonstrated that the approach achieved a high accuracy in occluded virtual scenarios and a promising result in real indoor scenarios, with increased performance when considering higher error thresholds. The obtained results show the potential of this system for use in AMRs to enhance the efficiency and safety of automated pallet handling in industrial settings in the future.

1 INTRODUCTION

Mobile robotics has repeatedly revolutionized the industry, reshaping the technology in material handling. As industries strive for greater efficiency and automation, warehouses and other sectors rapidly transition from traditional Automated Guided Vehicles (AGVs) to Autonomous Mobile Robots (AMRs), marking a paradigm shift in how tasks are executed, and goods are managed within dynamic environments (Fragapane et al., 2021). As those AMRs become an essential part of the warehouses, the importance of sophisticated perception capabilities, such as multi-instance 6D object pose estimation, becomes increasingly evident. It enables the robots to classify and recognize objects, estimate their poses, and track them over a period of time (Chen and Guhl, 2018). In this context, multi-instance 6D object pose estimation comprises the detection of objects and an estimation of their 3D translation and 3D rotation. For some detection methods, this is a single stage, while others perform object detection and pose estimation as distinct stages (Gorschlüter et al., 2022). For the detection method, several techniques and algorithms exist to date having point clouds, RGB, Depth, or RGB-D images as their inputs, however, all of them face difficult challenges (robustness of detection) in the industrial environment such as noise and occlusions in the sensor data. Industrial environments, like factories and warehouses, often exhibit a cluttered arrangement of objects and machinery. This poses a challenge for accurate object detection using 6D methods, as it can be intricate to discern the object of interest. Also, dealing with objects that have different types of textures and symmetries can affect the performance of the AMRs. In some cases, the object may lack adequate texture or distinctive features, making it more difficult to accurately estimate the object's pose. Estimating the object's pose in real-time can also be a challenge in industrial environments due to the large amount of data that needs to be processed.

In this work, a multimodal approach for detecting and estimating the 6D pose of pallets within an industrial environment is proposed. The goal with this detection system is, in the future, to integrate it into an autonomous forklift platform. This autonomous forklift will be capable of navigating to designated drop-off and pick-up zones, detecting pallets of interest, and managing their transportation. The integration of this system into the production line is expected to significantly enhance the warehouse's efficiency and reinforce workplace safety by further improving the automation of forklift maneuvers.

The proposed multimodal detection approach is based on the DenseFusion framework (Wang et al., 2019), with changes introduced in the feature fu-

Lourenço, J., Arsénio, G., Garrote, L. and Nunes, U.
Multimodal 6D Detection of Industrial Pallets, in Real and Virtual Environments, with Applications in Industrial AMRs.
DOI: 10.5220/0013073300003822
Paper published under CC license (CC BY-NC-ND 4.0)
In *Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics (ICINCO 2024) - Volume 2*, pages 345-352
ISBN: 978-989-758-717-7; ISSN: 2184-2809
Proceedings Copyright © 2024 by SCITEPRESS – Science and Technology Publications, Lda.

sion and geometry feature extraction stages. Due to difficulties on acquiring data on running factories with an AMR, and to not hinder the research on this topic, we also prepared three datasets. Two datasets were acquired and annotated in a virtual environment, containing multiple industrial shelving units and pallets, while one dataset was acquired indoor and user-annotated, with a pallet in different positions and with different levels of occlusion.

2 RELATED WORK

The problem of 6D object detection is widely studied in different fields of robotics. It pertains to the process of recognizing 3D objects within a 3D space and determining their positioning (X, Y, Z) and orientation (roll, pitch, yaw). The different approaches to this problem can be divided into RGB-based approaches and RGB-D-based approaches.

RGB-based approaches can be holistic or based on the dense correspondence. Holistic approaches involve directly extracting the pose parameters from RGB images, as in the case of DeepIM (Li et al., 2020), a method that takes as an input the initial 6D pose estimation of an object in the image and outputs a relative SE(3) transformation that is compared to the initial pose to improve the estimate. On the other hand, dense correspondence approaches establish correspondences between image pixels and mesh vertices to recover poses using Perspective-n-Point (PnP) techniques, like in the Coordinates-Based Disentangles Pose Network (CDPN) (Li et al., 2019) that separates the pose estimation process into distinct predictions for rotation and translation.The rotation estimation employs a carefully designed local region-based framework, enhancing both accuracy and efficiency. For translation estimation, the network directly derives this information from localized image patches. These distinct tasks are integrated and addressed within a single unified network. Given that the size of an object in an image can vary significantly with its distance from the camera, the object is scaled to a fixed size based on the detection output. Finally, another approach is 2D-keypoint based that detect 2D keypoints to establish the 2D-3D correspondence for pose estimation, although they may suffer from loss of geometry information due to perspective projections. This is the case of the Pixel-wise Voting Network (Peng et al., 2022) which employs regression on pixel-wise vectors to infer the positions of keypoints, which are subsequently utilized to cast votes for keypoint localization. This methodology establishes a versatile representation capable of accurately localizing keypoints, even in scenarios where they may be occluded or truncated. Furthermore, this approach provides a means to assess the uncertainties associated with keypoint locations, thus offering valuable insights for the PnP solver.

Due to RGB-D images being easy to obtain, RGB-D based approaches are widely investigated in the problem of 6D object detection. They can be divided into different methods such as template-based methods, that rely on feature and shape-based template matching to locate the object in the image and roughly estimate its pose, such as (Cao et al., 2016), which employed a 3D model to generate example poses of a textureless object to identify the closest match to the input image using GPU implementation. Their method involved transforming images into the Laplacian of the Gaussian space to ensure invariance to changes in illumination and appearance. To enable real-time matching, the authors proposed modifications to the template set and the image, as well as a restructuration of the conventional normalized cross-correlation operation. These adjustments allowed for the harnessing of the computational power of the GPU to perform rapid matrix-matrix multiplication. Feature-based methods are also used in this type of approaches, they exploit the point cloud to match 3D features and fit the object models into the scene. The approach proposed by (Hinterstoisser et al., 2016) in 2016 with a series of enhancements to the PPF approach (Drost et al., 2010). These advancements encompass sampling and voting schemes aimed at mitigating the influence of clutter and sensor noise. The sampling scheme selects pairs of points that are probable to belong to the same object, while deliberately avoiding pairs considered likely to belong to different objects on the background. The voting scheme then consolidates the PPF of all pairs of points anticipated to belong to the same object, while disregarding those anticipated to belong to different objects or the background. Finally, Deep-Learning-based methods like DenseFusion (Wang et al., 2019) which processes separately RGB and depth in two main stages. First it processes the inputted color images to perform semantic segmentation for each object, and then processes the results of the segmentation and estimates the object's 6D pose using an iterative pose refinement module that increases the precision of orientation estimation with a small inference time.

3 METHODOLOGY

The pipeline of the proposed framework illustrated in Figure 1, which uses RGB and Depth as inputs, is

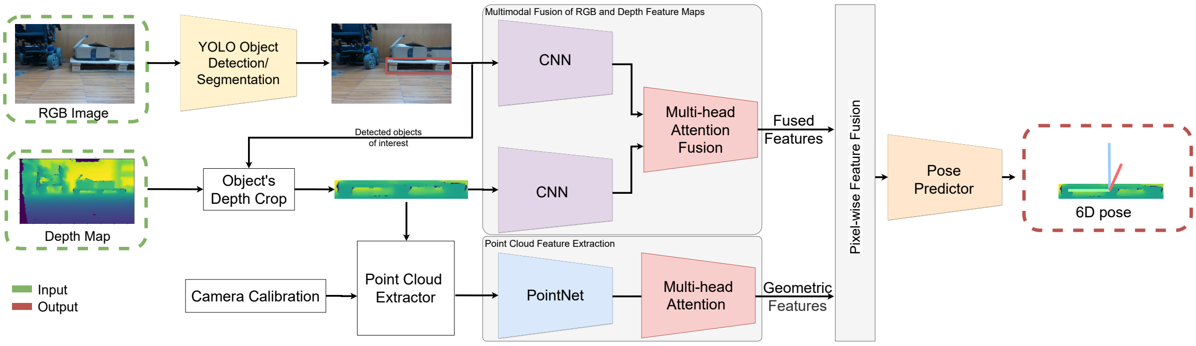
RGB Image
YOLO Object Detection/ Segmentation
Detected objects of interest
Depth Map
Object's Depth Crop
Camera Calibration
Point Cloud Extractor
Multimodal Fusion of RGB and Depth Feature Maps
CNN
CNN
Multi-head Attention Fusion
Fused Features
Point Cloud Feature Extraction
PointNet
Multi-head Attention
Geometric Features
Pixel-wise Feature Fusion
Pose Predictor
6D pose
🟩 Input
🟥 Output

Figure 1: Pipeline of the proposed framework using RGB and Depth for pallet detection and 6D pose prediction.

heavily inspired on the DenseFusion approach (Wang et al., 2019) and reuses several of its modules, including the RGB feature extraction network, the point cloud feature extraction network, the pixel-wise feature fusion, and the pose predictor. Modifications on the input of the pipeline to use object detection instead of a segmentation network and multi-head self-attention at different levels are introduced to improve over the DenseFusion approach, creating a new approach that can be deployed in an AMR considering a shared object detection system.

3.1 Object Detector

The initial stage involves the RGB-D information as input and performs an object detection for each object of interest in the image. The object detection network is the YOLOv8 network (Jocher et al., 2023), composed of a backbone network, a neck network and a head network. The backbone network is built upon a custom CSP-Darknet53 network (Wang et al., 2020) and has a Spatial Pyramid Pooling Fast (SPPF) layer. The neck network employs a Path Aggregation Network (PANet) structure, which helps the model to effectively capture features at several scales by flowing information across different spatial resolutions. Finally, the head network is responsible for generating the final outputs, such as bounding boxes and confidence scores for each object. For each frame processed in the object detector, a set of bounding box detections is obtained. For each detection that contains a pallet, a crop of the RGB and Depth images is performed considering the bounding box shape, to guarantee that only the object's shape and texture is processed in the subsequent steps, one object at a time.

3.1.1 RGB and Depth Feature Extractors

The RGB feature extractor is a modified version of the Residual Network (He et al., 2016) integrated with the Scene Parsing Network (Zhao et al., 2017) module. The main goal of the feature extractor is to get relevant features from RGB images. The backbone of the feature extractor is the ResNet, known for its ability to train deep models effectively. The ResNet network uses residual blocks that allow the network to learn residual functions, which represent the difference between the input and output of a layer in a neural network, instead of unreferenced ones. This means that instead of attempting to learn the complete identity mapping from the initial stages, the network can focus on learning the changes, or ”residuals,” to the input's identity mapping. The residual block consists of two or three streams of convolutional neural networks, followed by an element-wise addition operation that combines the input with the output of the convolutional layers.

In this particular implementation, ResNet-18 serves as the backbone. The different ResNet architectures vary in the number of layers, therefore in this case it consists of 18 layers. ResNet-18 is the optimal option when weighing the trade-off between accuracy and computing resource usage.

The PSPNet module is incorporated to enhance the feature extractor's capability to extract quality features. The PSPNet module utilizes a pyramid pooling strategy to capture multiscale contextual information from the input image. The feature maps are divided into multiple stages, each employing adaptive average pooling and convolution operations to extract features at different spatial resolutions. The original features are then concatenated with these features after bilinearly upsampled (a process of increasing the spatial resolution of feature maps using bilinear interpolation, which estimates new pixel values based on the linear interpolation of neighboring pixels). To improve efficiency, a bottleneck convolutional layer minimizes the dimensionality of the concatenated features. The process of feature extraction starts with the RGB image passing through the ResNet backbone. Then, the network extracts both low-level and high-level features from the image. These features are then processed by the PSPNet module, which captures contextual information at multiple pyramid scales and incorporates it into the feature representation. By incorporating this module, object pose estimation is now robust to scale variations, making it unaffected by changes in scale. For the depth feature extraction, the first layer of the ResNet was modified to process the 1-channel depth image.

3.1.2 Multimodal Fusion of RGB and Depth Feature Maps

This stage involves selecting which features from the RGB (F_{RGB}) and depth (F_D) are relevant to estimate/predict the object's 6D pose. A multi-head self-attention (Srinivas et al., 2021) strategy is employed in order to capture the most relevant features from both modalities.

Attention mechanisms (Luong et al., 2015) provide the network with salient features from each modality, which minimizes the noise and irrelevant information. This approach enables the network to decide when and how to integrate RGB and Depth data. These mechanisms generate attention weights that emphasize the most salient features from each modality.

Given the availability of both RGB and depth features, the introduction of an attention mechanism aims to fuse the two modalities to leverage complementary information. Let $F_{RGB} \in \mathbb{R}^{d_{RGB}}$ represent the feature vector derived from the RGB modality, where d_{RGB} is the dimensionality of the RGB feature space, and $F_D \in \mathbb{R}^{d_D}$ represent the corresponding depth features, where d_D is the dimensionality of the depth feature space. To fuse the two modalities, we concatenate these feature vectors along the feature dimension:

$$F_F = [F_{RGB}, F_D] \in \mathbb{R}^{(d_{RGB}+d_D)} \tag{1}$$

The combined feature vector F_F contains information from both RGB and depth modalities for each spatial location. Next, to model the relevant feature interdependencies and relationships, we employ the multi-head self-attention mechanism. The multi-head self-attention mechanism allows each location to attend to all other locations, enabling the model to capture relevant feature interactions. The attention mechanism computes a weighted sum of all the feature representations, where the weights are determined dynamically based on the similarity between the query and key vectors. For each attention head, the query (Q), key (K), and value (V) matrices are computed from the combined feature representation F_F:

$$Q = W_Q F_F, \quad K = W_K F_F, \quad V = W_V F_F \tag{2}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{(d_{RGB}+d_D)\times d_{head}}$ are learned projection matrices and d_{head} is the dimensionality of each attention head. The attention weights are computed as:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_{head}}}\right)V \tag{3}$$

The outputs from multiple attention heads are concatenated and projected back to the original feature space, yielding a more comprehensive representation. By applying multi-head self-attention, the model captures both spatial and cross-modal interactions between the RGB and depth features, leading to a richer representation that combines both appearance and depth characteristics.

3.1.3 Point Cloud Feature Extraction

From the object's cropped representation, and using the camera's intrinsic parameters, a 3D point cloud is obtained. From the set of 3D points from the point cloud, a N_{PC} number of points is selected (P). If a mask of the object is present, the points are selected within the exported points that represent the object, otherwise they are uniformly sampled without repetition. If the 3D point cloud's size is below N_{PC}, the point cloud is oversampled to match the N_{PC} points. This can occur for objects that are too occluded or that are far away from the camera, but its pose estimate is still required. The point cloud feature extraction employed is derived from the DenseFusion's implementation, as it uses a PointNet-like architecture to extract per-point geometric features. An additional multi-head self-attention mechanism is introduced at the end, similarly to the approach presented in Section 3.1.2, to focus the network on the geometric features more relevant to the 6D pose estimation task.

3.1.4 Pixel-Wise Dense Feature Fusion Network

The objective of the Pixel-wise Dense Feature Fusion Network is to fuse the information obtained from the image and the 3D point cloud. The concept behind the pixel-wise dense fusion network is to move away from relying solely on the object's global features to determine its pose. Instead, the DenseFusion approach performs local per-pixel fusion so that it is possible to make predictions based on each feature. In more practical terms, to each point of the point cloud P a set of features are associated, composed of global features, geometric features and fused features. The global features are common to each point $p \in P$, and are obtained from a Multi-Layer Perceptron (MLP) using all geometric features and fused features as inputs. This process aims to minimize the effects of occlusion and detection/segmentation noise. This allows the method to select the most reliable representations based on the visible portion of the object, reducing the impact of issues such as objects partially hidden from view or interference from background elements.

3.1.5 Pose Estimator

The pose estimator block in the Dense Fusion architecture estimates the 6D pose of known objects from the RGB-D images. The block takes the pixel-wise dense feature embedding from the Pixel-wise Dense Feature Fusion network as input and outputs the predicted pose of the object. The fused features are processed using an MLP which outputs a 3D vector representing the translation of the object in the 3D space, a quaternion representing the rotation of the object and a confidence coefficient that represents the quality of the pose estimate. This block uses a residual-based approach to estimate the pose, and the pose estimation loss is calculated by measuring the distance between the observed object's point cloud (P) and the corresponding object's points centered on the object's center of mass (P^M) transformed by the estimated pose (T). This loss is quantified by the distance by those points and is defined as:

$$L = \frac{1}{N_{PC}} \sum_{N_{PC}}^{i=1} (|T p_i^M - p_i| c_i - w \; log(c_i)), \qquad (4)$$

where c_i is the confidence coefficient, w is a balancing hyperparameter used as a secondary regularization term to balance the average distance loss and confidence, and p and p^M are points from the sets of points P and P^M respectively.

The network's output comprises N_{PC} point predictions. Each prediction includes the rotation quaternion, translation vector, and confidence coefficient, all contributing to the estimated pose. By incorporating the confidence coefficient, the network can autonomously evaluate the quality of its predictions. The object's 6D pose prediction is the one associated with the highest confidence coefficient.

4 EXPERIMENTAL VALIDATION

To validate the proposed framework, datasets tailored to the needs of the 6D pose estimation problem for pallets were needed. In particular, due to the absence of realistic and readily available datasets online for validating the accuracy of the detection of pallets within an indoor or warehouse setting. Datasets such as the PalLoc6D dataset (Knitt et al., 2022), which serves as an RGB-D virtual dataset for the 6D detection of pallets, lack a realistic scenario because the pallets are generated randomly in various locations, surrounded by random objects, within a randomized background. Since the dataset introduces unrealistic backgrounds that do not represent real scenarios that an AMR may find, in this work we propose two virtual datasets considering pallets in an industrial shelve. Additionally, to validate the proposed method in a real scenario, a small indoor dataset was also acquired.

4.1 Evaluation Datasets

This section presents a detailed explanation of the three datasets created to evaluate the proposed pipeline; two datasets generated in a virtual dataset and one dataset in an indoor setting. Samples from the three datasets are shown in Fig. 2.

4.1.1 Virtual Pallet Dataset

The first virtual dataset was created due to the lack of a realistic and available online dataset to validate the accuracy of the detection of pallets within a warehouse setting. The key idea revolves around an AMR such as a forklift capable of navigating towards designated pick-up and drop-off zones. Once positioned correctly, the robot must accurately identify the pallet's location, enabling seamless execution of the loading and unloading processes. To achieve this objective, the dataset simulates a virtual warehouse environment (see Fig. 3), consisting of carefully designed shelves populated with pallets and boxes, capturing data from the perspective of a robotic forklift. The acquisition process is automated from a set of predefined camera positions. The virtual dataset that was produced comprises 816 RGB-D raw images with a resolution of 1224x370 coupled with the corresponding point clouds, 2D and 3D bounding box annotations for every pallet object within the image, as well as the essential calibration matrices. Additionally, the system exports the masks of the objects in the scene, focusing only in this context on the pallets.

4.1.2 Virtual Pallet Dataset with Occlusions

The second dataset was acquired on the same scenario as the previous one (see Fig. 3), and introduces occlusions to make it closer to the reality in industrial environments. This dataset tries to simulate the scenarios where the AMR is not completely aligned with the pallets during the pick-up process. The acquisition process is automated from a set of predefined camera positions, but a noise factor is introduced to create misaligned and occluded views. Different pallet locations were added, along with scenarios where the pallet was barely visible due to occlusions from boxes or

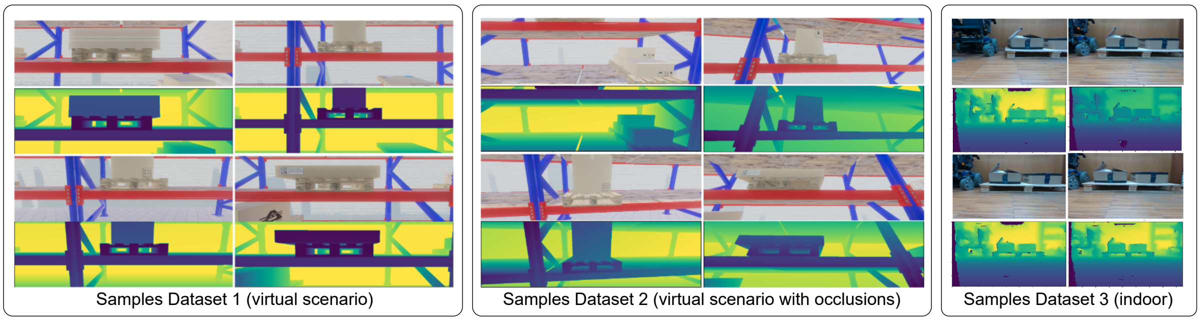
Samples Dataset 1 (virtual scenario)
Samples Dataset 2 (virtual scenario with occlusions)
Samples Dataset 3 (indoor)

Figure 2: Sample RGB and Depth images, of the three evaluation datasets.

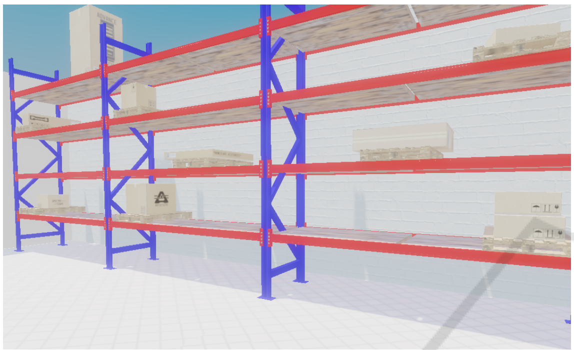

Figure 3: Virtual environment developed to acquire the virtual datasets.

other elements of the warehouse. The virtual dataset that was produced comprises 1632 RGB-D raw images with a resolution of 1224x370 coupled with the corresponding point clouds, 2D and 3D bounding box annotations for every pallet object within the image, as well as the essential calibration matrices.

4.1.3 Indoor Pallet Dataset

In order to make a first approach from simulation to reality, an indoor dataset comprised of, 1597 RGB-D images were generated. We used a mobile robot equipped with an Intel RealSense D435 camera. This choice was based on its affordability and the high-quality RGB sensor, which is capable of producing excellent images even in low-light conditions. The depth sensor performs well within a range of 2 to 4 meters, but its accuracy diminishes for more distant objects, likely due to the low-light environment. A scenario with a real pallet was created, with multiple boxes stacked over the pallet. The robot would move close to the pallet and then a box would be removed, and the run replicated, until the pallet was empty. A final run was included without the pallet to serve as additional background. The RGB-D images were acquired in ROS, exported and processed in the Roboflow platform (Dwyer et al., 2024). Using the Roboflow interface, the pallets were annotated and the dataset created. Its interface supports various annotation types, including bounding boxes, polygons, and key points, allowing for precise delineation of objects within images. In the context of this work, Roboflow was used to label the pallets in the collected 2D RGB images, preparing them for further processing and analysis.

After the labelling process, aided by the Roboflow interface, an in-house software was used to crop the labels and assign a 6D pose to each detection using the point cloud obtained from the depth image (using the camera's intrinsic parameters).

4.2 Experimental Results

This section presents the performance and results of the proposed approach. A brief explanation about the evaluation metrics will be conducted, afterwards the validation in each dataset will have a distance-based accuracy study, to evaluate the network's ability to estimate object poses at various distances from the sensor, as well as a multimodal study to analyze how different input data can impact a model's performance.

4.2.1 Evaluation Metric

The evaluation of the method's performance will be presented in terms of Average Distance of Keypoints (ADD). The ADD is first referenced by Hinterstoisser *et al.* (Hinterstoisser et al., 2012) and is a metric that computes the average Euclidean distance between the estimated keypoint positions ($\hat{R}$, $\hat{t}$) and the ground truth pose positions. The lower score indicates a greater accuracy of the pose estimation algorithm, and it is computed as follows:

$$ADD = \frac{1}{N_{PC}} \sum_{p \in P} ||(p - (\hat{R}p^M + \hat{t})||, \tag{5}$$

where $\hat{R}$ is the estimated rotation, and $\hat{t}$ is the estimated translation, p represents one of the sampled points belonging to the point cloud P and p^M the corresponding point of the object with its ground-truth

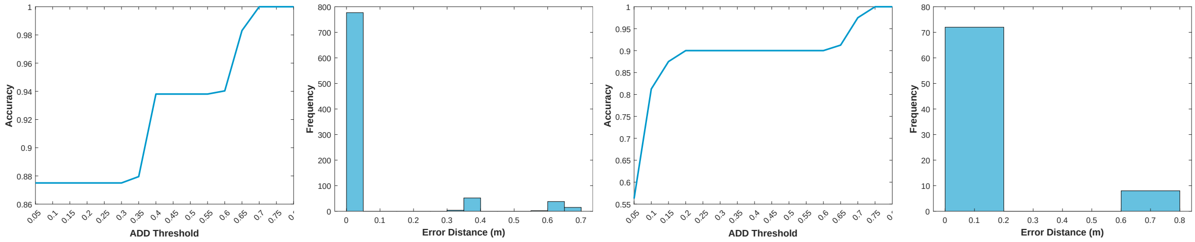

Figure 4: Accuracy ADD according to the distance of the objects and distribution histogram of ADD per pallet instance, for the virtual and indoor datasets, respectively.

rotation and translation removed.This metric can effectively function as both a loss function and a measure of accuracy. Predictions that attain a score lower than a predetermined threshold are considered correct.

4.2.2 Virtual Pallet Dataset Performance

Before introducing and analyzing the results obtained, it is important to note that initial validation tests were performed in the LINEMOD dataset (Hinterstoisser et al., 2013), although the dataset context is out of scope from the topic of pallet detection it is still relevant to point out that in our tests a baseline DenseFusion architecture achieved 95.3% on average accuracy using the LINEMOD dataset thresholds, and our approach achieved 98.1% on the same conditions. It is important to also note that these results were obtained with the inclusion of a refinement step that we do not include in this work, as it did not improve the results in both virtual and indoor datasets.

The results on the first virtual dataset were a performance of 100% considering a threshold on the ADD of 0.05, it is important to reinforce that this dataset represents the ideal conditions an AMR may observe, and as such a high to perfect accuracy is expected. For the second virtual dataset, the obtained performance was ±88%. The results for different ADD thresholds are shown in Fig. 4. The x axis represent the ADD threshold, and the y axis represent the accuracy obtained. To better assess the results of the method, Fig. 4 shows the distribution of the ADD distance per pallet instance and in this case the majority of the pose estimations had an error distance inferior to 0.1 meters, leaving the remaining ADD clusters at ± 0.4 and ± 0.7. From an analysis of the data, these clusters correspond to overly occluded pallets where only a small number of points were extracted, and an oversampling strategy was employed. In the future, such objects may be automatically rejected as its 6D pose is difficult to predict. Overall, the results demonstrate that the proposed framework is capable of achieving high accuracy, even in occluded scenarios, as shown by both the accuracy curve and the ADD distribution histogram.

4.2.3 Indoor Dataset Performance

On the indoor dataset, the obtained performance was ± 56% for a similar ADD threshold of 0.05 meters. Figure 4 shows the accuracy with relative ADD threshold distance. For a different threshold of 0.1 the method's performance rises to ± 82%. This may be caused due to the noisy nature of the real data, that was affected by motion artifacts as well as by the poor performance of the depth sensor due to varying luminosity. When the analysis focuses over the ADD distribution, it reveals that the majority of estimated poses have an error close or inferior to 0.1 meters. The ADD cluster on the 0.7 meters represents a similar behavior as observed on the second virtual dataset. In particular, the accuracy curve shows a similar trend as in the virtual dataset, but with slightly different results. The curve starts at around ± 56% accuracy for an ADD distance of 0.05 meters, indicating that for very small errors, the accuracy is lower when compared to the virtual dataset. The accuracy improves significantly as the error distance increases and if we set an acceptable 0.1-0.2 meters of error distance, due to incorrect annotation (as it was performed in the Depth generated point cloud), or due to small occlusions, we achieve an accuracy between 80 and 90%. The ADD distribution histogram shows that the majority of poses have an error distance of less than 0.1 meters, indicating that the framework is able to estimate most object poses with high precision.

5 CONCLUSIONS

This work presents a multimodal 6D pose estimation of industrial objects in real and virtual environments, particularly aimed at future integration with AMRs. Using the DenseFusion framework as a basis, an enhanced version is proposed combining RGB and Depth and utilizing multi-head self-attention mechanisms for robust feature fusion. The method was tested on two virtual datasets, including scenarios with occlusions, and a real-world indoor dataset, showing promising results even under challenging conditions such as occlusions and noise. The proposed framework achieved, expectedly, a better accuracy in the occluded virtual dataset than on the real-world indoor dataset, due to the noisy nature of the measurements (that is not replicated in the virtual datasets). Still, these results demonstrate the potential of the approach for future applications in industrial environments, where it can significantly enhance efficiency and safety. Future work will include the acquisition of a new dataset in an industrial setting, with further validation of the method proposed.

ACKNOWLEDGEMENTS

This work has been supported by the Portuguese Foundation for Science and Technology (FCT) through grant UIDB/00048/2020 (DOI 10.54499/UIDB/00048/2020) and by Agenda “GreenAuto: Green innovation for the Automotive Industry”, with reference PRR-C644867037-00000013.

REFERENCES

Cao, Z., Sheikh, Y., and Banerjee, N. K. (2016). Real-time scalable 6DOF pose estimation for textureless objects. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*.

Chen, X. and Guhl, J. (2018). Industrial Robot Control with Object Recognition based on Deep Learning. *Procedia CIRP*, 76:149–154.

Drost, B., Ulrich, M., Navab, N., and Ilic, S. (2010). Model globally, match locally: Efficient and robust 3D object recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Dwyer, B., Nelson, J., Hansen, T., et al. (2024). Roboflow (version 1.0) [software]. https://roboflow.com.

Fragapane, G., De Koster, R., Sgarbossa, F., and Strandhagen, J. O. (2021). Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda. *European Journal of Operational Research*, 294(2):405–426.

Gorschlüter, F., Rojtberg, P., and Pöllabauer, T. (2022). A Survey of 6D Object Detection Based on 3D Models for Industrial Applications. *Journal of Imaging*, 8(3):53.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA. IEEE.

Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., and Navab, N. (2013). Model Based Training, Detection and Pose Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes. In *11th Asian Conference on Computer Vision*. Springer.

Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Konolige, K., Bradski, G., and Navab, N. (2012). Technical Demonstration on Model Based Training, Detection and Pose Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes. In *Computer Vision – ECCV 2012. Workshops and Demonstrations*, volume 7585. Springer Berlin Heidelberg, Berlin, Heidelberg.

Hinterstoisser, S., Lepetit, V., Rajkumar, N., and Konolige, K. (2016). Going Further with Point Pair Features. volume 9907, pages 834–848. arXiv:1711.04061 [cs].

Jocher, G., Chaurasia, A., and Qiu, J. (2023). Ultralytics yolov8. https://github.com/ultralytics/ultralytics. Accessed: 2024-06-5.

Knitt, M., Schyga, J., Adamanov, A., Hinckeldeyn, J., and Kreutzfeldt, J. (2022). PalLoc6D-Estimating the Pose of a Euro Pallet with an RGB Camera based on Synthetic Training Data. https://doi.org/10.15480/336.4470.

Li, Y., Wang, G., Ji, X., Xiang, Y., and Fox, D. (2020). DeepIM: Deep Iterative Matching for 6D Pose Estimation. *International Journal of Computer Vision*, 128(3):657–678. arXiv:1804.00175 [cs].

Li, Z., Wang, G., and Ji, X. (2019). CDPN: Coordinates-Based Disentangled Pose Network for Real-Time RGB-Based 6-DoF Object Pose Estimation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Peng, S., Zhou, X., Liu, Y., Lin, H., Huang, Q., and Bao, H. (2022). PVNet: Pixel-Wise Voting Network for 6DoF Object Pose Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6):3212–3223.

Srinivas, A., Lin, T., Parmar, N., Shlens, J., Abbeel, P., and Vaswani, A. (2021). Bottleneck transformers for visual recognition. *CoRR*, abs/2101.11605.

Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., and Savarese, S. (2019). DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion. arXiv:1901.04780 [cs].

Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020). CSPNet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid Scene Parsing Network. arXiv:1612.01105 [cs].