A Comparative Assessment of Parcel Box Detection Algorithms for Industrial Applications

Ernesto Fontana[1], William Zarotti[1] and Dario Lodi Rizzini[1]

***Abstract*—Industrial logistics may benefit from object perception to perform flexible and efficient management of goods. This paper illustrates and experimentally compares two approaches to parcel box detection in depth images for an industrial depalletization task. The model-based method detects clusters in the input point cloud according to curvature and other geometric features, and aggregates the candidate objects. The learning-based method relies on the state-of-the-art Mask R-CNN, which has been retrained on an acquired dataset with missing measurements. The target object poses are evaluated through standard geometric registration. The experiments on acquired datasets show the feasibility of the two approaches.**

I. Introduction

An important task of industrial logistics is handling products and arranging them into pallets. Parcel boxes or bottle bundles are palletized for efficient storage and shipping at the end of production pipelines [1]. In order to adapt to customer orders, the stored pallets may be depalletized and arranged into the requested pallet format. Sensor-driven object detection enables manipulation of products in arbitrary configurations.

RGB-D cameras are relatively affordable sensors, which acquire both the appearance and metric data of an observed scene. However, industrial settings pose diverse challenges: varying illumination conditions, low lights, low-texture and symmetric objects, clutters, self-occlusions, viewpoint limitations due to robot motion constraints. Object manipulation often relies on the accurate and robust estimation of the pose. The problem is usually addressed in two steps. First, the system searches one or more areas of the depth image corresponding to the searched object. Then, pose is estimated, for example by aligning a template with the measurements obtained from detection. Model-based algorithms address this problem by finding geometric features and patterns characterizing standard goods like parcel boxes. Machine manufacturers and developers generally appreciate the reliability and control of such approach.

The rise of deep learning methods in computer vision gives an alternative and effective solution to image segmentation and object detection. Although originally developed and trained in different contexts, some of these algorithms can easily adapt to novel classes of objects, including those commonly used in industrial logistics. While they satisfactorily address object detection, end-to-end estimation of pose is not accurate enough for manipulation tasks [2] and must be performed through geometric reasoning.

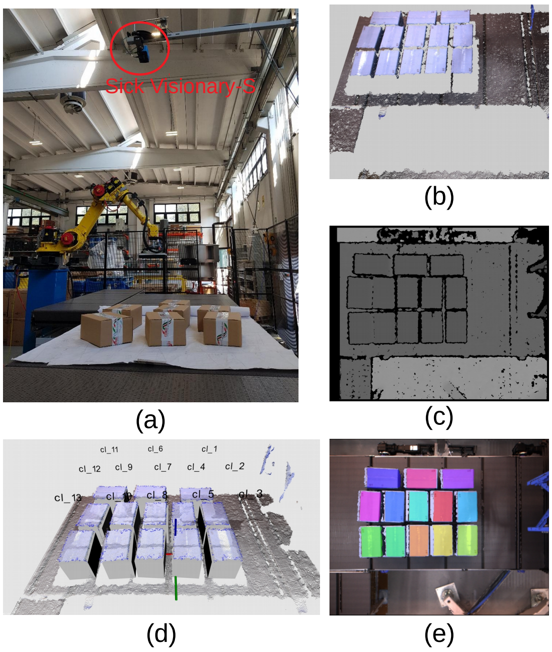

Figure 1: Outline of the system. (a) Experimental setup with the boxes on the depalletizer conveyor belt observed by Sick Visionary-S depth camera. (b) Colored point cloud and (c) depth image acquired by the camera displaying missing data. (d) The boxes detected with the model-based algorithm with the pose aligned box models (overlapped boxes in gray). (e) Boxes detected using the custom trained Mask R-CNN algorithm.

The contribution of this paper is the comparative investigation of model-based and learning-based approaches to object detection and pose estimation for an industrial depalletizing application. The industrial depth camera *Sick Visionary-S* captures the top view of the parcel boxes taken from an arbitrary pallet layer deposited on a conveyor belt (Figure 1(a)). The point clouds and depth images (Figure 1(b)-(c)) suffer from missing measurements and altered colors. The acquired data are elaborated following two different approaches, which we have labeled as model-based and, respectively, learning-based. The model-based method performs an initial segmentation finding Locally Convex Connected Patches (LCCP) [3], and groups the connected supervoxels into clusters, according to their similarity in curvature and color. The poses of the boxes are then obtained through registration between each cluster and the box geometric model (Figure 1(d)).

[1]Department of Engineering and Architecture, University of Parma, Italy. {ernesto.fontana, dario.lodirizzini}@unipr.it william.zarotti@studenti.unipr.it

978-1-6654-1213-1/21/$31.00 ©2021 IEEE

The learning-based algorithm is based on Mask R-CNN [4], [5] re-trained for generic box detection. Training set consists both of a general purpose parcel box dataset, based on free images, and of a specific dataset acquired in an industrial setup. The major issue found in the images acquired by Sick Visionary-S is with invalid range measurements, and the corresponding missing image pixels. Although the original Mask R-CNN has been designed for, and trained on complete RGB images, it has proven adaptable to the new application after proper training (Figure 1(e)). Pose estimation is still addressed through registration, with customized operations on the found clusters.

Comparative experiments have assessed the performance in object detection and pose estimation of the two approaches. The already mentioned experimental dataset has been acquired by mounting the depth camera on an industrial depalletizer and observing the parcel boxes on the conveyor belt. The samples consist of arbitrary images for detection experiments and box configurations with known relative poses. The achieved results illustrate the feasibility of both the approaches.

II. Related Work

Computer vision and sensor technology have been used for a long time in automation (e.g. in quality control inspection), but their extensive use for end-line manipulation and logistics is more recent. An early work [6] presents a genetic algorithm to detect and label parcel boxes arranged in multi-layer pallets using a gray scale camera. Prasse et al. [7] illustrate a method for pallet load detection using a time-of-flight (ToF) sensor and RFIDs. More recently, some model-based methods [8], [9] have been proposed for depalletization using depth sensors with or without color data.

Deep learning methods have also been applied to detection and manipulation of products. The winning team of Amazon Picking Challenge 2017 [10] used a Fully Convolutional Network to recognize objects with different shapes. While a notable effort, the contest setting, as well as the variety of the scene, do not entirely reflect the usual industrial setup and requirements. Because of this, the achieved results are deeply influenced by the choice of an appropriate dataset. Significant efforts have been made to learn both object detection and poses, from color or depth images [11], [2]. Furthermore, the systems developed in research projects like ILIAD [12] establish structured procedures in product manipulation driven by deep learning. Object-RPE [13] combines Mask R-CNN and multi-view registration to achieve recognition and evaluation of several object poses.

III. System Overview

Depalletization is the process of unloading pallets by taking their component boxes one by one. The depalletizing machine presented here decomposes the pallet by layers, bringing the current top layer to a conveyor belt using a large gripper. Then, the boxes are mechanically separated through friction, so that they picked up by one or more industrial robotic manipulators, to arrange them into a new pallet layout, according to customer orders. In general, the layout of the input layer is arbitrary and can vary. The dimensions of the boxes are assumed parameters of the algorithm, although this hypothesis may not be used. As discussed, the palletizing machine separates the boxes, but there is no guarantee about the achievable pairwise object distances. Thus, the algorithms must operate also objects in contact.

In our system, a Sick Visionary-S depth camera is mounted on pole and observes the goods directly from above, at about $1.80\ m$ height from the conveyor belt, as shown in Figure 1(a)-(c). Such depth camera combines stereo vision processing and structured light patterns in order to acquire accurate and reliable measurements. It returns 640×512 depth and RGB images, as well as point clouds, covering a $60^\circ \times 50^\circ$ field-of-view, and up to $2.5\ m$ range. The sensor is calibrated by the manufacturer, and the data fusion algorithm is not accessible. Unfortunately, the returned RGB image is not directly the one acquired by one of the two cameras; it is instead the virtual image associated to the depth measurements. The pixels corresponding to invalid range measurements have invalid color marked as black, as shown in Figure 1. These missing data represent an issue for any image processing, including the deep learning algorithms designed for complete data. The Sick Visionary-S returns data w.r.t. an arbitrary reference frame, which could be placed on the conveyor belt. Such frame transformation can also be handled externally from the camera processor, which is convenient. The goal of the system is to detect the boxes and evaluate their position and orientation w.r.t. the frame used by the sensor. The estimation must be performed after one single image, as the boxes move on the conveyor belt and the manipulators have to sequentially pick each of them. The admissible position error is about $5\ mm$, while the angular one is about 1°.

IV. Model-based Method

A. Detection

Model-based approaches include a range of solutions that vary from keypoint feature extraction in images or point clouds [14] to detection of geometric patterns. The parcel boxes to be recognized are generally patternless objects characterized by orthogonal planar faces. The Sick Visionary-S sensor described in section III provides reliable range data, whereas the images are affected by holes due to missing measurements. Moreover, it is fixed directly above the objects and captures their top faces. Additional working hypotheses are that the estimation is performed using a single acquisition and that no relative distance among the boxes should be assumed, i.e. they can be contiguous, even though the depalletizer tends to separate the products.

The initial step performs background removal, having the goal to eliminate the conveyor belt plane, on which lie the parcel boxes, from the acquired scene. In many commercial solutions, segmentation is trivially obtained by searching range discontinuities w.r.t. the background. The limitations of such naive technique lie in potential inaccuracies due to missing data, in mismanagement of contiguous target objects, and in

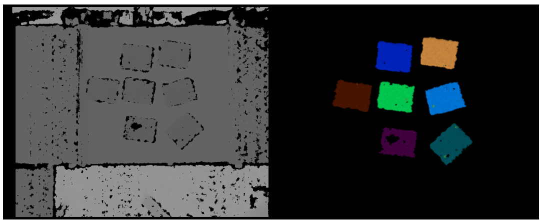

Figure 2: An example of model-based detection. Each color identifies a different cluster representing a box.

the inability to handle unexpected configurations (e.g. boxes not lying on the ground due to unpredictable deposit of the pallet layer on the machine).

We implemented a more general and robust solution based on LCCP algorithm [3]. This algorithm involves the segmentation of the input point cloud $\mathcal{C}_{in}$ into clusters $\{\mathcal{V}_i\}$. Internally, LCCP computes a finer supervoxel subdivision according to normal direction and connectivity. These supervoxels are merged based on their reciprocal curvature and other similarities, including the color of the points. Voxel resolution is the main parameter for the decomposition, while curvature smoothness threshold is decisive for the supervoxel clusterization. An example of the results of the described detection method is illustrated in Figure 2. LCCP output is a set of labeled clusters of points.

Although the industrial depalletizer described in section III is designed to separate the boxes in order to facilitate manipulation, we implemented a solution to recognize a group of continuous boxes, that are detected as a single cluster. Such solution requires the knowledge of the boxes' dimensions. The points of the cloud are handled as planar points corresponding to their projection on the ground and classified as foreground (object) or background (scene) points. First, the contours of connected components of foreground region are extracted and represented by a closed polygon. Since the clusters may have ”holes”, due to invalid range measurements, there may be internal polygon rings that can be easily removed. Then, the algorithm finds the orthogonal vertices of the external ring of the polygon, i.e. polygon vertices with angle close to 90°. The vertices are used to place the boxes on the contour through exhaustive search. The best box configuration is the one maximizing the overlap area between the boxes and the cluster polygon.

While this procedure has been implemented and tested, the palletizing machine used in this section is able to separate the boxes of the input pallet layers by using consecutive conveyor belts at different speeds, manipulation and other mechanisms. For these reasons, the examples presented in the experiments section consist of separate boxes.

B. Pose Estimation

Once the clusters corresponding to candidate objects are found, the algorithm evaluates their poses w.r.t. the sensor, through registration between each cluster of points and the templates of the boxes. Additional assumptions are assumed to achieve accurate estimation. The strongest one is the knowledge of box dimensions, that allows the definition of the target object template. The algorithm also requires that each cluster corresponds to a different box. The detection algorithm described in the previous section returns a collection of clusters $\{\mathcal{V}_i\}$, each corresponding to a single box. Pose estimation is addressed as a 2D registration problem, since the boxes lie on the conveyor belt, and their allowed motion is on the plane. The algorithm presented in the following can be easily generalized to estimate 3D poses (e.g. using a 3D template box and Procrustes algorithm), but the palletizing machine requires planar position and orientation. Furthermore, the point clouds acquired by the depth camera provide a top view of the scene, with few points on the lateral faces of the boxes.

Algorithm 1 Pose Estimation of a Box

```
function PoseEstimation($\mathcal{C}$, $w$, $h$)
    // Compute initial guess of poses
    $\boldsymbol{\mu} \leftarrow (\sum_{j=1}^{n} \mathbf{p}_j)/n$;
    $\boldsymbol{\Sigma} \leftarrow (\sum_{j=1}^{n} (\mathbf{p}_j - \boldsymbol{\mu})(\mathbf{p}_j - \boldsymbol{\mu})^\top)/n$;
    $\mathbf{v}_{max} \leftarrow \mathit{eigenvector}(\boldsymbol{\Sigma})$;
    $\theta \leftarrow \mathrm{atan2}\,(\mathbf{v}_{max,y}, \mathbf{v}_{max,x})$;
    $\mathbf{T} \leftarrow \mathrm{transformMatrix}(\boldsymbol{\mu}, \theta)$;
    // Template box border of size $w \times h$
    $\mathcal{B} \leftarrow \mathrm{box}(w, h)$;
    while !stopping do
        // Associate points to box
        for $p_j \in \mathcal{C}$ do
            $\mathbf{q}_j \leftarrow \mathrm{argmin}_{\mathbf{q} \in \mathcal{B}} \|\mathbf{T}^{-1}\mathbf{p}_j - \mathbf{q}\|$;
            $\mathbf{q}_j \leftarrow \mathbf{T}\ \mathbf{q}_j$;
        end for
        // New Estimation of pose
        $\mathbf{T} \leftarrow \mathrm{solveProcrustes}(\{(\mathbf{p}_j, \mathbf{q}_j)\}_{j=1,\ldots,n})$;
    end while
end function
```

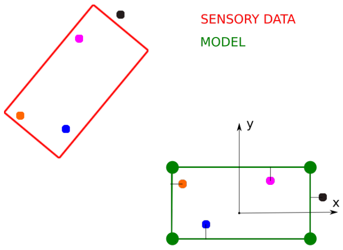

Figure 3: Data association between points and box contour.

Algorithm 1 illustrates the pose estimation algorithm. The input data of the algorithm are the point cloud $\mathcal{C}$, the width w and height h of the box. The point cloud $\mathcal{C}$ corresponds to the contour points of each cluster $\mathcal{V}_i$, i.e. only the border points. A virtual template box $\mathcal{B}$, i.e. a rectangle shape, is built with the given dimensions and centered in the reference frame origin. The goal is to find the transformation matrix $\mathbf{T}$ that better aligns $\mathcal{B}$ to $\mathcal{C}$. First, an initial guess of the pose of the algorithm is computed using the centroid $\boldsymbol{\mu}$ of the cloud for translation, and the eigenvectors of covariance $\boldsymbol{\Sigma}_i$ for orientation. Then, the pose is refined according to iterative closest point (ICP) algorithm. Each point $\mathbf{p}_j$ of the cluster is associated to its closest point $\mathbf{q}_j$ of the box template $\mathcal{B}$. To find the closest point on the box, it is easier to transform the point $\mathbf{p}_j$ to the box $\mathcal{B}$ centered on the origin and then re-transform the found $\mathbf{q}_j$ (line 13). The association is iteratively updated after each new estimate of transformation $\mathbf{T}$. The association is based on a "closest point in contour" metric, graphically illustrated in figure 3.

The output of the association is a list of pairs $(\mathbf{p}_j, \mathbf{q}_j)$, $\mathbf{p}_j$ belonging to the detected cluster and $\mathbf{q}_j$ its closest point on the virtual box template. The transformation minimizing the average distance between these pairs is the solution of the Procrustes problem. The Procrustes problem is solved iteratively until stopping conditions are reached, producing as output one affine transformation matrix, which is the geometric element that contains all the information needed for the pose estimation.

V. Learning-based Method

A. Detection

The learning-based method consists of two modules itself. The first module solves the instance segmentation problem for parcel box detection through the Mask R-CNN network. The second module operates on the segments of the point cloud corresponding to the boxes, in order to estimate their poses. This two step procedure exploits both the presented state-of-the-art deep learning algorithm for instance segmentation, and the efficient geometric estimation limited to a region of interest (ROI), modeled as a 2D Object-Oriented Bounding Box (OOBB) containing a single box.

The detection phase is carried out entirely by the Mask R-CNN neural network. The output of the network consists of the masks corresponding to the objects to be recognized. The network is distributed in pre-trained form for specific object categories, which do not include the parcel boxes. Thus, we initially trained it on a dataset of 1144 RGB images containing boxes with different sizes, colors and patterns in different contexts and environments. Since the upper faces of these boxes are rectangular, each mask is summarized with its OOBB, which is then used to label the dataset. Mask R-CNN has the capability to adapt and detect generic objects with a relatively limited new training set. This means that objects like parcel boxes, that have a well defined geometry, can be easily recognized.

The images acquired by the Sick Visionary-S suffer from missing pixels and color distortion, as discussed in section III. At this point, it is important to note that Mask R-CNN can output insufficiently precise identification of the edges, which would lead to inaccurate estimation of object orientation. As a matter of fact, orientation is mainly computed by using the points that are close to the edges, which are often missing due to invalid range reading. Hence, the decision of capturing an additional dataset of 220 images by mounting the depth camera on the palletizer has been made. Adding these images to the training set caused the network to estimate the rotation more precisely, especially for the intended application proposed in this paper. This dataset has been labeled using an expanded OOBB around each box. The expanded OOBB is computed by enlarging the rectangle from its center along its longest and shortest sides by 8 mm. Unfortunately, the expanded OOBBs of different boxes may intersect each other potentially causing misclassification. The impact of this issue on the pose estimation algorithm is explained next.

B. Pose Estimation

Pose estimation could be performed using the technique presented in section IV-B. Due to the particular and previously discussed issues, we preferred to implement a customized procedure. Following [15], the mask returned by the neural network selects not only the ROI of the target box in the image, but also in the corresponding point cloud. Then, pose estimation is performed using only the subset of points corresponding to the ROI.

The limited visibility of the lateral faces of boxes, along with the constrained motion on the conveyor belt, make 3D pose estimation superfluous and even error prone. The pose of each object is almost completely defined by its planar position and orientation w.r.t. the conveyor belt plane. Hence, the primary goals are the extraction of points of the upper face, alongside the removal of all points from the adjacent boxes, conveyor belt, soil or side faces of the box itself, that have been included in the expanded OOBB. To achieve these, we assume that upper face points all have approximately the same height from the ground plane, and that the OOBB center lies entirely on it.

First, the algorithm coarsely filters the points having the same height as the center, with a $4\ cm$ tolerance error. Each ROI returned by Mask R-CNN, and expanded as described above, may also include points belonging to other boxes. Connected components search discards the parts belonging to the other boxes. The operation is carried out after projecting all the points on the conveyor belt plane and according to planar adjacency criterion. This operation can output multiple rectangle-like clusters, which are all discarded except the one having the largest perimeter. A further refinement is performed on the remaining points by only keeping the ones whose height is consistent with the average height of the cluster, assuming a reduced tolerance range of $1\ cm$. The goal is to better remove the points lying on the lateral faces. Finally, the box pose is computed as the pose of the minimum rectangle containing the points.

VI. Experiments

Experiments, as well as dataset acquisitions, have been performed on the depalletizing system illustrated in section III. The Sick Visionary-S depth camera observes the scene directly from above, returning a top view of the depalletizer and the boxes. The acquired observations are in the form of depth images, RGB images and colored point clouds. The exposure time has been set according to the lighting conditions.

We acquired two datasets. The dataset *MOVE* consists of about 551 images of parcel boxes in arbitrary poses moved by the conveyor belt. Each image contains on average about

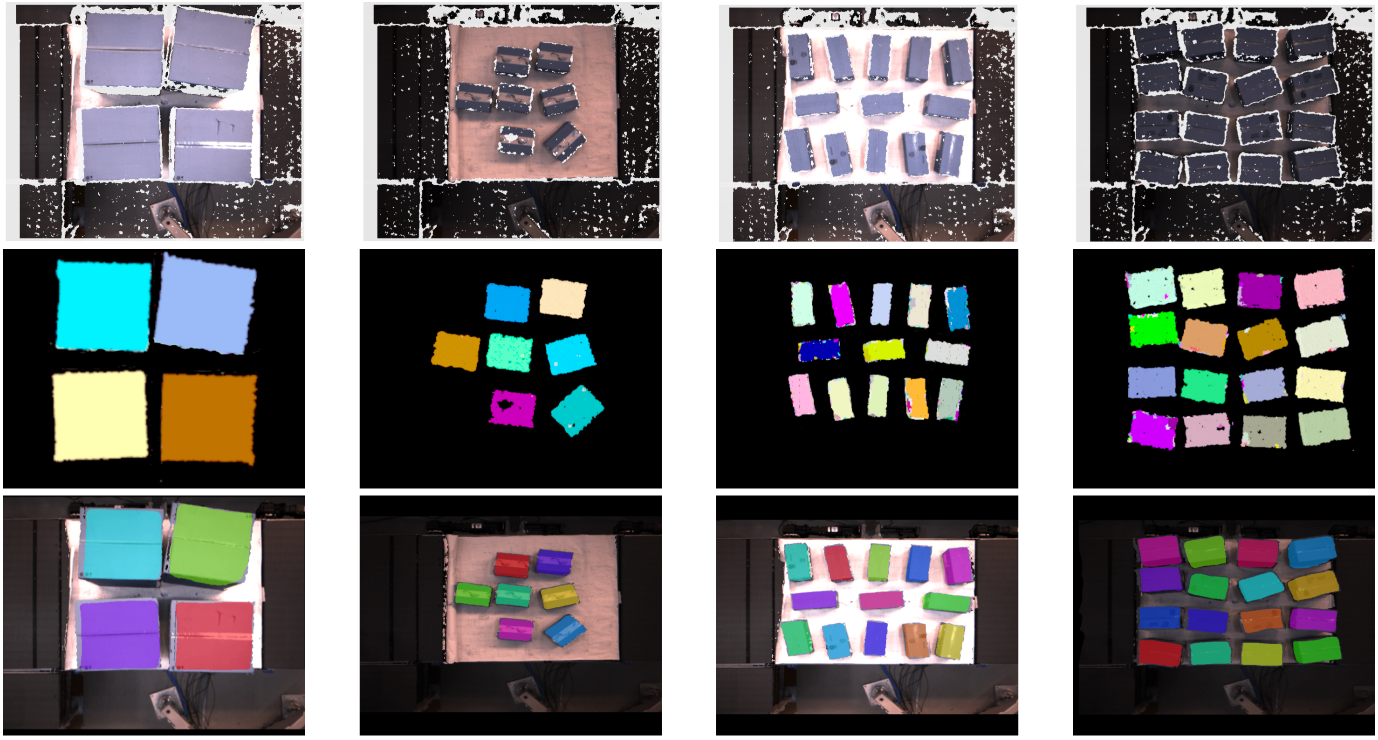

Figure 4: Detection results comparison. First row: input RGB depth image; second row: model-based results; third row: learning-based results.

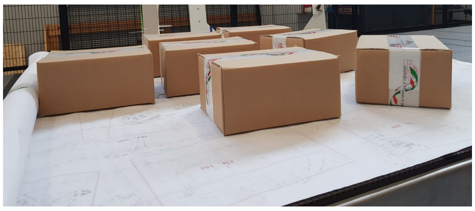

Figure 5: The paper board with the printed configurations of the parcel boxes used as groundtruth.

13 boxes. The main use of this dataset is to provide different configurations for training the Mask R-CNN. As discussed in section V, 220 images have been labeled and used as training set. The dataset *POSE* consists of 610 images of static parcel boxes. We used four parcel box formats that are shown in the first row of Figure 4, presenting different detection challenges, in terms of box size and patterns. The parcel boxes have been arranged according to the configurations printed on a paper board (Figure 5). Since the relative poses marked on the board are known, these configurations are used as groundtruth in pose accuracy experiments. The four configurations are labeled as *box-4*, *box-7*, *box-13* and *box-16* according to the number of boxes appearing in each of them. The dataset also includes single box acquisitions and other preliminary captures on arbitrary box configurations, which have not been used.

Both the proposed model-based and the learning-based methods correctly detect the parcel boxes in all the images or clouds. Figure 4 illustrates some detection results of the model-based (second row) and learning-based (third row). Since it operates on the point cloud according to geometric criterion, the first algorithm detects the top face of the box, while Mask R-CNN finds the full mask of the boxes in the RGB image.

	model-based method		learning-based method	
	pos_err[mm]	ang_err[deg]	pos_err[mm]	ang_err[deg]
box-4	3.4	0.87	2.9	0.82
box-7	5.2	1.62	5.6	1.19
box-13	6.1	1.33	6.0	0.83
box-16	6.4	1.54	7.3	1.19

Table I: Position and orientation errors of the algorithms.

The final goal of the perception system is to estimate the poses of the parcel boxes w.r.t. the sensor frame, and then transform them into the machine frame, since this is what is required by the depalletizer in order to manipulate the boxes. Table I returns the quantitative results of pose accuracy tests on dataset *POSE*. The position and angular errors reported in this table are the average errors obtained by comparison between the estimated box poses and the groundtruth poses reported on the paper board. The groundtruth poses have been rigidly aligned with the estimated poses and then compared. The position errors have the tendency to become larger when more boxes are present in the scene, possibly due to cumulative errors, and also to the imperfect manual placement of boxes on the board. However, the errors are close to the target position and angular errors of about 5 mm and 1°. The errors are comparable for

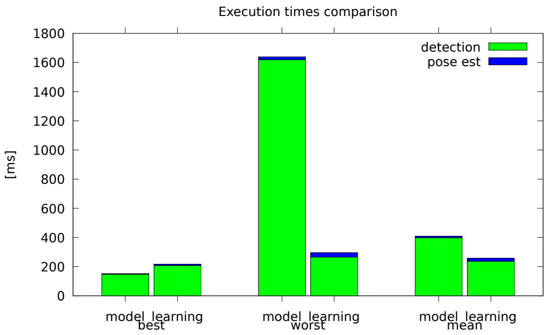

Figure 6: Best, worst and mean execution times of the proposed model-based and learning-based algorithms.

the two methods. The learning-based method achieves slightly better results in the simpler configurations, whereas the model-based method performing becomes comparatively better when the number of boxes increases.

The execution times of the two algorithms are shown in Figure 6. The computation is dominated by the detection of the parcel boxes w.r.t. pose estimation. Unfortunately, it has not been possible to execute all the tests on the same computer. The model-based algorithm has been performed on a machine equipped with *Intel Core i5-7200U* processor, 8 GB RAM and a 2 GB *NVIDIA GeForce 920M* GPU. The learning-based approach has been performed on an *Intel Core i5-4210U@1.70GHz* CPU, alongside a dedicated 4 GB *NVIDIA GeForce GTX 970* GPU. Under these operating conditions, the average execution time of the learning-based algorithm is less than the time required by the model-based algorithm. The execution of the model-based algorithm is also less predictable as suggested by the maximum time, possibly due to the number of clusters found by LCCP. However, this assessment does not lead to absolute conclusion. Indeed, the choice of the GPU hugely affects the performance of Mask R-CNN: the same algorithm executed on a different machine equipped with GPU NVIDIA GeForce 840M takes on average 2000 ms and up to 4500 ms.

VII. Conclusion

This paper has presented and compared a model-based and a learning-based method for object detection and pose estimation using a depth camera in industrial depalletization. The first approach exploits the geometry of the standard shape of the products to segment the input point cloud and perform pose estimation. The second one adopts the state-of-the-art Mask R-CNN for instance segmentation and registration to evaluate boxes position and orientation. Although the neural network has been originally trained for different contexts without images with missing data, it has proven as adequate as the customized model-based method. Both algorithms effectively detect the targets and estimate their pose with compliant accuracy to the application requirements. A machine learning algorithm like Mask R-CNN has proven an off-the-shelf reliable solution also for industrial applications. Its main drawback lies in the integration effort required to run learning-based systems on current industrial technologies, more targeted to reliability rather than computational effort. Moreover, further investigation is required to assess the amount of training for adopting more radical context changes and to test other neural network architectures. The experiments performed in this work suggest the potential of this approach for industrial applications.

Acknowledgment

This work is supported by project "MAF Macchine Autonome & Flessibili", MISE_FCS_DM 5 marzo 2018

References

[1] W. Echelmeyer, A. Kirchheim, and E. Wellbrock, "Robotics-logistics: Challenges for automation of logistic processes," in *2008 IEEE International Conference on Automation and Logistics*, Sep. 2008, pp. 2099–2103.

[2] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "DeepIM: Deep Iterative Matching for 6D Pose Estimation," *Int. Journal of Computer Vision*, vol. 128, pp. 657–678, 2020.

[3] C. Stein, M. Schoeler, J. Papon, and F. Woergoetter, "Object partitioning using local convexity," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *CoRR*, vol. abs/1703.06870, 2017. [Online]. Available: http://arxiv.org/abs/1703.06870

[5] M. Runz, M. Buffier, and L. Agapito, "Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects," in *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2018, pp. 10–20.

[6] M. Hashimoto and K. Sumi, "Genetic labeling and its application to depalletizing robot vision," in *Proceedings of IEEE Workshop on Applications of Computer Vision*, Dec 1994, pp. 177–186.

[7] C. Prasse, S. Skibinski, F. Weichert, J. Stenzel, H. Müller, and M. ten Hompel, "Concept of automated load detection for de-palletizing using depth images and RFID data," in *IEEE Int. Conf. on Control System, Computing and Engineering*, 2011, pp. 249–254.

[8] P. Arpenti, R. Caccavale, G. Paduano, G. Andrea Fontanelli, V. Lippiello, L. Villani, and B. Siciliano, "Rgb-d recognition and localization of cases for robotic depalletizing in supermarkets," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 4, pp. 6233–6238, 2020.

[9] D. Chiaravalli, G. Palli, R. Monica, J. Aleotti, and D. Lodi Rizzini, "Integration of a Multi-Camera Vision System and Admittance Control for Robotic Industrial Depalletizing," in *IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*, Sept. 2020, pp. 667–674.

[10] A. Zeng, K.-T. Yu, S. Song, D. Suo, E. Walker, A. Rodriguez, and J. Xiao, "Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2017.

[11] Y. Xiang, S. Tanner, V. Narayanan, and D. Fox, "PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes," in *Proc. of Robotics: Science and Systems (RSS)*, 2018.

[12] T. Stoyanov and M. M. (ORU), "Safe feature-based navigation for industrial agvs," in *ILIAD (Intra-Logistics with Integrated Automatic Deployment: Safe and Scalable Fleets in Shared Spaces) Deliverable 6.4: Perception system for detecting boxes and wrapping*, 2018, pp. 1–29.

[13] D.-C. Hoang, A. J. Lilienthal, and T. Stoyanov, "Object-RPE: Dense 3d reconstruction and pose estimation with convolutional neural networks," *Robotics and Autonomous Systems*, p. 103632, 2020.

[14] A. Aldoma, Z. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. Rusu, S. Gedikli, and M. Vincze, "Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 80–91, Sept. 2012.

[15] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3d object detection from rgb-d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 918–927.