IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 August 2023, accepted 12 September 2023, date of publication 12 October 2023, date of current version 18 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3324058

APPLIED RESEARCH

A Hybrid CNN-DSP Algorithm for Package Detection in Distance Maps

ELENA VASILEVA, (Graduate Student Member, IEEE), AND ZORAN A. IVANOVSKI, (Senior Member, IEEE)
Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje, 1000 Skopje, North Macedonia

Corresponding author: Zoran A. Ivanovski (mars@feit.ukim.edu.mk)

This work was supported by Fast Global Solutions, Inc., and Mikrosam AD Prilep, through the scope of the project “Object recognition in 3D scenes based on depth and image data.”

ABSTRACT This paper presents a hybrid algorithm for real-time instance segmentation of packages from scenes represented by 2D distance maps (range images). The paper introduces a novel approach combining deep learning-based methods and digital signal processing methods to enable accurate package recognition, using a small training dataset with high variability and distance measurement errors characteristic of Time-of-Flight-based scanning. Two convolutional neural networks with architecture optimized for training with a limited number of samples perform an initial segmentation of package components (sides and edges). An algorithm based on digital signal processing methods performs refinement of intermediate results, and combines package components into packages. Training and evaluation of the algorithm were performed on a custom dataset containing scenes of packages, shipping bags, and packaging of irregular shapes with various sizes, orientations, and degrees of occlusion, organized either in ordered stacks or arbitrary order. The convolutional neural networks provide a reliable distinction between components of packages and components of other types of packaging and surroundings. Package sides containing a sufficient number of distance points are correctly combined into packages. Thus, the proposed algorithm represents a solid basis for fully automated loading/unloading of packages with arbitrary sizes and materials from transport trailers and storage spaces. The dataset and annotations for box side surfaces are available at: https://dipteam.feit.ukim.edu.mk/results-package-detection.html.

INDEX TERMS Automated loading/unloading, CNN, depth maps, edge detection, instance segmentation, package recognition, planar surface detection, semantic segmentation.

I. INTRODUCTION

Loading/unloading of packages with arbitrary sizes and packaging materials from transport trailers or storage spaces is predominantly a fully manual process, and there are no existing fully automated systems to date. Successful partial or total process automation of loading/unloading of packages will significantly increase the cost-effectiveness of the loading/unloading process by reducing the time needed and the number of damaged goods. Since cardboard packages of various sizes and materials are the most common type of packaging used for transporting goods, fast and correct package segmentation is the basis for constructing automated systems for trailer loading/ unloading. The main steps towards package segmentation are precise detection and localization of packages of different sizes, shapes, orientations, and varying degrees of occlusion, ordered in both organized and unorganized piles; and successfully differentiating packages from other types of packaging, such as bags and irregular objects (cylindrical packaging etc.).

The package loading systems on the market are designed to operate on packaging types with a strictly defined set of different sizes, orientations, and materials, in a heavily controlled environment. This limits their use to specific use cases (shoe boxes [1], evenly ordered cardboard packages of predefined sizes [2], [3], and plain, non-reflective

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

cardboard packages [4], [5]). Some systems use additional input information to localize and identify the packages (QR codes [2], etc.). The details of the vision algorithms used by these systems are not publicly available. Previous research on general pick-and-place systems is also scarce, and usually considers only a small variety of objects in a controlled environment [6], [7], [8]. This results in a lack of comprehensive labeled datasets necessary for successfully training end-to-end deep learning algorithms, due to which most of the algorithms employ only DSP-based (Digital Signal Processing) techniques. The lack of comprehensive datasets forces broadening the research area and breaking the object detection problem into multiple consecutive steps.

Packages can be made of materials with different textures, colors and reflectivity, which results in very different representations in color photographs. However, all shipping packages are cuboid boxes of variable sizes, and these features are clearly distinguishable in a depth representation of the scene. Package detection systems operating on distance images are independent of color and texture. However, such systems must adapt to a wide variety of distance measurement errors due to imperfections of depth scanning systems. Based on this, we propose a data-driven hybrid algorithm for package detection in scenes represented with distance maps.

The proposed algorithm is a novel integration of CNNs (Convolutional Neural Networks) and DSP-based methods applied to the problem of segmenting rectangular packages with arbitrary sizes, orientations, and degrees of occlusion, from piles of packages in arbitrary order. A custom dataset of packaging scenes represented by 2D distance maps containing heavy surface distortion represents the training and evaluation set for the initial deep learning-based segmentation of package components (sides and edges). The box side and edge detection are both performed by end-to-end trainable CNN models with a small number of parameters, trained on data with heavily imbalanced classes. Custom algorithms for region expansion and edge thinning refine the box sides and edges. The segmented sides and edges provide the input for the rule-based box forming algorithm which combines the box sides to form complete boxes.

The rest of this paper is organized as follows. An overview of previous works regarding segmentation of geometric shapes and related problems is given in Section II. Section III covers the general overview of the package detection algorithm. A description of the custom dataset and the CNNs for initial segmentation of package sides and edges are presented in Section IV. Section V details the box forming algorithm. The experimental results and discussion are presented in Section VI, and finally, conclusions are presented in Section VII.

II. RELATED WORK

A. DETECTION OF DIFFERENT PACKAGING TYPES

The topic of automated package loading/unloading is discussed in few previous works. Proposed algorithms and systems use range data from laser scanners for recognizing and localizing goods inside shipping containers or precisely ordered onto pellets. The proposed solutions rely heavily on modeling techniques to represent the 3D shape of different types of packaging, and are usually restricted to limited size ranges and orientations. A method for package segmentation based on superquadric segmentation in range images is presented in [6]. The algorithm uses fusion of region and boundary information to segment rectangular packages in scenes containing a small number of packages (at most five) with a low degree of occlusion. A modeling approach for segmenting heavy, inflexible sacks stacked into ordered rows is given in [7]. The range data acquired by a 2D laser scanner is segmented based on the predefined characteristics of the sack surface. The paper elaborates on the difficulties in creating models for specific cases, such as tunnels or overlapping sacks. A complete robot vision system for detection and localization of objects with different shapes and sizes in piled packaging scenarios is presented in [8]. In this paper, an initial segmentation of object parts in point cloud data is based on the geometric properties of boxes, bags, and barrels, followed by additional geometric criteria to combine the segmented parts into complete objects. Different approaches based on topology graphs and Gauss maps for segmenting different types of objects in point clouds for various target usages are presented in [9] and [10], respectively. Reference [11] presents a modeling-based method for corner detection in low-resolution 2D range images, drawing attention to false positive detections of objects with rounded and irregular shapes, and analyzing the drop in accuracy caused by reducing the number of different range levels. Detection and localization of a large cuboid-shaped container using connected component analysis and model fitting are presented in [12]. The algorithm requires that there is a minimum surrounding free space around the object. Initial object localization is performed by detecting the container wheels in a 2D representation of the scene, and verification of the container orientation is performed by fitting a simple box model to a 3D point cloud obtained with a 3D ToF (Time-of-Flight) camera. A model fitting algorithm for detecting payload in the form of pellets, followed by estimating the position of the pellets in a frontal view, is presented in [13]. Weichert et al. [14] propose a similar approach to detect box-shaped payloads on euro pellets in point clouds obtained using 3D ToF cameras, and present successful results for objects represented with a sufficient number of points. An algorithm for detecting cardboard packages from RGB-D images by fitting cuboid models to detected box faces is described in [15]. The RGB-D camera is mounted onto a robotic arm, enabling optimal positioning of the camera to get the clearest view of the scene at every approach. Non-reflective package surfaces represented with a sufficient number of points are successfully detected. Other works presenting detection and localization of objects in 3D point clouds using machine learning techniques can be found in the literature [17], [18]. These works show promising results, at the expense of creating and annotating large databases to successfully train the networks.

As discussed in the previous works, correct segmentation of packages from scenes containing payloads of arbitrary shape, size and orientation, in a cluttered environment with a high degree of occlusion, requires an extremely large dataset to represent all possible configurations and adapt to the wide variety of distance measurement errors characteristic of the distance measurement technologies (LIDAR, ToF IR range scanners, etc.). Most of the published works performing detection of packaging types focus on using modeling techniques based on DSP methods, and omit deep learning solutions due to the unavailability of large comprehensive datasets. On the other hand, the traditional DSP methods have a large number of prerequisites, and provide limited success. Therefore, decomposing the packaging detection problem into consecutive straightforward tasks is the most often chosen approach.

In recent years, deep learning-based algorithms have achieved remarkable success in object detection. The emergence of CNNs resulted in different object detectors operating on a single RGB image. The two-stage CNN-based architecture proposed by He et al. [19] performs object detection and instance segmentation in RGB images. Efficient architectures [20], [21] have outperformed two-stage detectors in terms of inference time. One-stage detectors have provided faster and more efficient object detection [22], [23], and instance segmentation [24], at the cost of reduced accuracy regarding small objects. Recent state-of-the-art models (Transformer Neural Networks [25], [26], Focal Modulation Networks [27]) employ complex architectures with a large number of parameters to successfully leverage the concept of attention for improving the detection accuracy of small and occluded objects, at the expense of inference time and large requirements for training data.

The increase in model complexity has also given rise to end-to-end models able to perform multiple complex tasks based only on RGB input. Works in the field of monocular depth estimation [28], [29] and monocular 3D object detection [30] have provided results comparable to state-of-the-art methods leveraging both RGB and depth data. Recent advances in the field of 6D pose estimation of objects from RGB images have provided exceptional results in different target applications, demonstrating the power of large learning-based models. Park et al. [31] proposed Pix2Pose, an auto-encoder-based pose estimation method that predicts 3D coordinates of objects without textured 3D models, which successfully handles occlusion by leveraging GANs (generative adversarial networks) to recover occluded parts. The end-to-end CNN presented by Zhang et al. [32] extends a 2D object detection pipeline with a pose estimation module to indirectly regress the image coordinates of the object's 3D vertices based on 2D detection results. Fan et al. [33] propose a novel approach which achieves state-of-the-art category-level 6D object pose estimation results with only RGB image input.

Several papers have proposed object detection algorithms for detecting various packaging types using only RGB input. A CNN-based algorithm for detecting KLT packaging units from a single color photograph, published by D'orr et al. [16], operates on one type of packaging unit and provides successful detection of the visible faces of packaging units precisely ordered in a uniform stack, with two visible sides, and no occlusions. Naumann et al. [34] developed an algorithm to reconstruct the 3D shape of individual parcels from a single RGB image, finding that although knowledge gained by training on synthetic data can be applied in the real world to a certain extent, reliable deployment in different real-world scenarios is still challenging. The system developed by Castaño Amoros et al. [35] includes a module for detecting and recognizing separate pallets that contain unassembled corrugated cardboard packaging from top-view RGB images.

Several conclusions can be drawn from the review of the datasets and models used in the discussed works in terms of the number of samples in the dataset, degree of variability, and model complexity. Firstly, object detection in RGB images as an active area of research has achieved outstanding results in recent years. Outstanding results in the field of monocular computer vision have been reported in high-complexity tasks, such as monocular 3D object detection and pose estimation. However, the ability to accurately represent various objects and conditions from different real-life scenarios is a result of utilizing large, complex model architectures with millions of parameters, which require large training datasets. In cases where there is a lack of large, comprehensive labeled datasets, the practical use of the algorithms is limited to a predefined set of different object types and configurations, and controlled environments. Furthermore, although synthetic data can be beneficial for representing certain aspects of real-world scenarios, deploying models trained exclusively on synthetic data in diverse real-world scenarios still poses a challenge. As a result, the crucial aspect of creating an effective model designed to perform well in real-life scenarios with minimal constraints is the selection of the model architecture.

As discussed, both papers proposing object detection algorithms in photographs and in distance maps can be found in the literature. The preference for the input type is based on the specific use case – the material of the packages, and most importantly, the amount of data available. Shipping packages can be made of very different materials, textures, and colors. Therefore, a large number of samples are needed to represent all types of packages in color photographs. In distance maps or depth images, all packages are represented as boxes consisting of adjacent perpendicular planar sides marked with a gradual change in distance/depth represented with gray levels. The packages have straight, continuous edges marked by changes in either distance/depth or direction of change of distance/depth. Furthermore, the distance representation of the scene retains complete information about the objects' geometric structure, which provides a straightforward way to differentiate shipping packages from other packaging types. The depth scanning methods are susceptible to various distance measurement errors, such as irregularly erroneous distance measurements of highly reflective surfaces, different measurement values for adjacent surfaces with a sharp difference in color, and different manifestations of surface distortion at object edges. However, despite the distance measurement errors, the simplicity and uniformity of the depth representation of packages and their constituent components (sides and edges) is a strong motivation to use distance maps as input to our proposed package recognition algorithm.

CNNs present state-of-the-art results in image processing problems in the presence of sufficient labeled data. However, for practical use, especially for complex problems lacking large labeled datasets, deep learning methods are often combined with DSP methods forming hybrid algorithms. One of the key benefits of this approach is the ability to leverage the strengths of both techniques: CNNs are highly effective at learning complex feature representations from data, while DSP methods provide computationally efficient signal processing and analysis, independently of the amount of data available. Recent papers have proposed hybrid approaches combining different deep learning-based algorithms and DSP methods, demonstrating promising results in different applications such as speech recognition, image analysis, biomedical signal and image processing, and time series forecasting. In these studies, DSP methods are often used for enhancing the CNN results through pre- and post-processing; and parts of DSP algorithms are replaced with deep learning-based algorithms to achieve higher accuracy. Lopac et al. [36] utilize time-frequency representations to generate the input for an ensemble of CNN classifiers to detect non-stationary gravitational-wave signals in high noise, demonstrating an improvement over a baseline CNN model operating directly on the source signals. Abdelhamid et al. [37] use DSP methods for extracting task-specific features, which are input into a CNN-LSTM (Long Short-Term Memory) hybrid model for emotion recognition in speech. Yadav et al. [38] propose a novel multi-scale fusion of features generated by a CNN and an improved Canny edge detection algorithm [39] for detection of bone fractures in X-ray images, achieving an increase in accuracy over a baseline CNN model. Furthermore, ensembles of learning-based models are proposed to extract information from 3D spatial data and time series data. Montaha et al. [40] propose a hybrid 3D CNN-LSTM model to classify brain tumor on 3D MRI scans, and Sajjad et al. [41] propose a hybrid learning-based model combining CNNs and GRUs (Gated Recurrent Units) to form a unified framework for predicting energy consumption. The small number of labeled data with high variability makes a hybrid CNN-DSP algorithm a more suitable approach as opposed to an end-to-end learning-based solution.

The goal of the proposed algorithm is package recognition in cluttered environments, using only a small custom dataset of distance maps obtained with ToF-based scanning technology. The small, highly variable dataset, and the simple representation of the basic package components, motivate decomposing the problem into several consecutive steps. The initial phase of segmenting package sides and package edges utilizes deep learning methods to model the object and measurement distortions with a single end-to-end network. In the second phase, the components are combined into complete packages by a rule-based algorithm that takes into account the box side orientation, adjacency and type of edge formed by a pair of sides.

In the following Sections II-B and II-C, we present a short overview of the related work in semantic and instance segmentation, and edge detection, as it represents the basis for segmenting package components.

B. SEGMENTATION OF PLANAR SURFACES

Segmenting planar surfaces from a distance map is a straightforward task in an ideal environment since a plane is represented by a surface with gradually changing distance values, as proved in several previous works (RANSAC [42], wavelet segmentation [43], region growing [8]). However, although the DSP methods are good at rejecting true outliers and random noise, the drawbacks of these methods introduce the need for complex time-intensive post-processing even in an ideal environment. For example, the inability to segment instances of connected surfaces – coplanar points belonging to disjoint components will be grouped by RANSAC or wavelet segmentation. Furthermore, the large variability of distance measurement errors is impossible to represent by a unified set of rules, and previous works [8], [11] prove that DSP methods provide limited success even with simulated datasets that lack the faults of the scanning technologies. As a result, we look into works utilizing deep learning methods for segmentation applied to various other segmentation problems.

Recent advances in CNN-based semantic and instance segmentation in color photographs [19] have produced state-of-the-art results. However, many state-of-the-art models designed for general object recognition have a complex architecture requiring thousands of training samples [19], [44], and thus are not suitable for training with a small dataset. Pre-training of CNNs on large datasets [45], [46] is a commonly used strategy, proven useful in improving the overall accuracy of CNNs when working with small datasets. However, positive results are achieved only when pre-training is performed with similar data and for similar problems; pre-training for a significantly different objective hinders network performance.

Due to the rising use of LIDAR, especially in the field of automated navigation, the most widely used format of distance data is unstructured point clouds. Many recent works presenting deep learning-based point cloud segmentation algorithms show satisfying results [47], [48], [49], [50]. However, handling point cloud input requires adding pre-processing modules to reformat the unstructured point cloud input data, resulting in large and complex CNN architectures unsuitable for training with small datasets.

Papers introducing medical image segmentation models [51], [52], [53], [54], [55] present a similar problem – binary semantic segmentation of single-channel image data. The medical image processing research field regularly encounters a lack of manually labeled data, and the medical image datasets generally contain a small number of samples. Therefore, research in the field of medical image segmentation focuses on designing machine learning models with a limited number of parameters that can be successfully trained on relatively small datasets. This makes such models a good starting point in defining a network structure for distance map segmentation with a limited number of training samples.

C. EDGE DETECTION

Edge detection in the context of package recognition should enable detecting edges that belong to packages, while disregarding edges belonging to different objects and artificial edges within package sides that result from depth measurement errors or physical deformation of the packages. Differentiating the package edges from other edges relies on the structural difference of the different edge types. Package edges are long, straight, and continuous, separating two areas with different directions or rates of change of surface depth; as opposed to edges of other types of packaging items that are short, broken, and less emphasized. Objects with uneven, irregular surfaces (e.g. bags) contain a large number of small edges within the object surface.

Classifying the edges into different types (package edges and edges belonging to other types of packaging) using traditional DSP methods for edge detection is very difficult. This creates the need for learning-based methods for reliable edge detection in the context of package recognition in distance maps. Many recent state-of-the-art works present learning-based edge [56], [57] and contour [58], [59], [60], [61] detection designed for and tested on color photographs. However, unlike distance maps, photographs contain more edges within the objects resulting from object texture and color variation, posing the need for complex models with a large number of training parameters and large training datasets. Therefore, algorithms designed to work on color photographs are not suitable for small datasets of distance maps.

CNNs with an encoder-decoder structure are an efficient and straightforward solution for edge detection in a single feed-forward step without additional pre-processing. U-Net [51] provides top results in the field of object segmentation in both photographs and medical images. However, due to the large number of parameters, it requires large datasets to be successfully trained. Our work focuses on segmentation of single-channel images with small convolutional networks suitable to be trained on a limited number of samples, which makes previous medical image segmentation works [51], [52] an ideal starting point. In this paper, we propose an encoder-decoder CNN with a limited number of parameters, but enough capacity to retain the crucial dataset features for correct segmentation of box edges. This CNN, presented in our previous work [62], is briefly described in Section IV-C.

III. PACKAGE DETECTION

The proposed package detection algorithm is an integration of CNNs for initial segmentation of object components, DSP methods for refinement of object components, and rule-based criteria for object forming. The algorithm, shown in Fig. 1, consists of 3 main steps marked with dotted line rectangles. First, initial segmentation of package sides and package edges is performed with two CNN-based algorithms. An end-to-end trainable CNN with a small number of parameters performs segmentation of box sides as connected components. The segmented box sides are then expanded through region growing-based surface expansion. Each connected component in the binary mask of box sides represents one side of a package. Another end-to-end trainable CNN [62] with a similar structure performs segmentation of box edges as connected components. A custom edge thinning algorithm based on non-maximum suppression generates binary masks of edges with 1 pixel thickness.

The edge masks are used to refine the surface masks. The edge detection network is given priority in this step due to providing higher accuracy in areas of rounded inner edges. Since the shape and orientation of the box sides is crucial for combining the segmented box sides into packages, perspective correction is applied to the raw distance maps, thereby generating a depth map of the scene.

Next, a rule-based box forming algorithm combines the package sides to form complete packages. Since a side directly facing the scanner is the only visible side of the package, we first detect the sides facing the scanner. The remaining package sides are combined into packages using a set of rules based on the geometric properties of packages in Euclidean space. The sets of sides where each pair of sides fulfills the given criteria form a package.

Finally, with surface expansion of the package sides and forming closed borders around each side, we obtain the final segmented packages. The input to the package detection algorithm is a raw distance map of 144 pixels × 176 pixels, where the pixel values represent the distance of the scene points to the scanner. Sections IV and V describe the package segmentation algorithm in detail.

IV. SEGMENTATION OF PACKAGE SIDES AND EDGES

A. DATASET

1) DATASET DESCRIPTION

The custom dataset used for training and evaluation of the proposed algorithm consists of distance maps representing different scenes of stacked packaging items of three types:

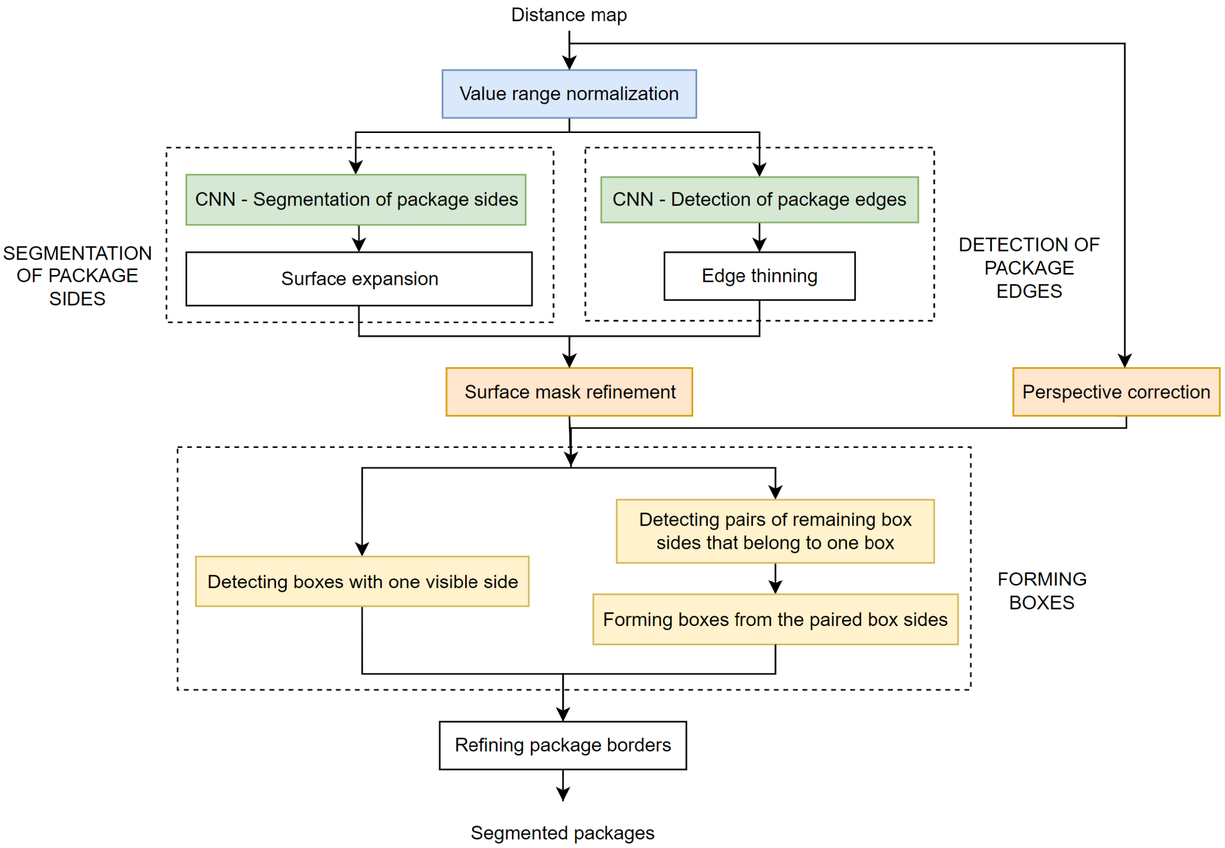
Distance map
Value range normalization
SEGMENTATION OF PACKAGE SIDES
CNN - Segmentation of package sides
Surface expansion
CNN - Detection of package edges
Edge thinning
DETECTION OF PACKAGE EDGES
Surface mask refinement
Perspective correction
Detecting pairs of remaining box sides that belong to one box
Detecting boxes with one visible side
Forming boxes from the paired box sides
FORMING BOXES
Refining package borders
Segmented packages

FIGURE 1. The proposed algorithm for package detection. The data preparation is marked in blue. The green blocks represent two deep convolutional neural networks. The orange blocks define the refining of CNN results and preparing the input for the package forming algorithm. The rule-based algorithm for forming packages from the segmented package sides is represented with yellow blocks.

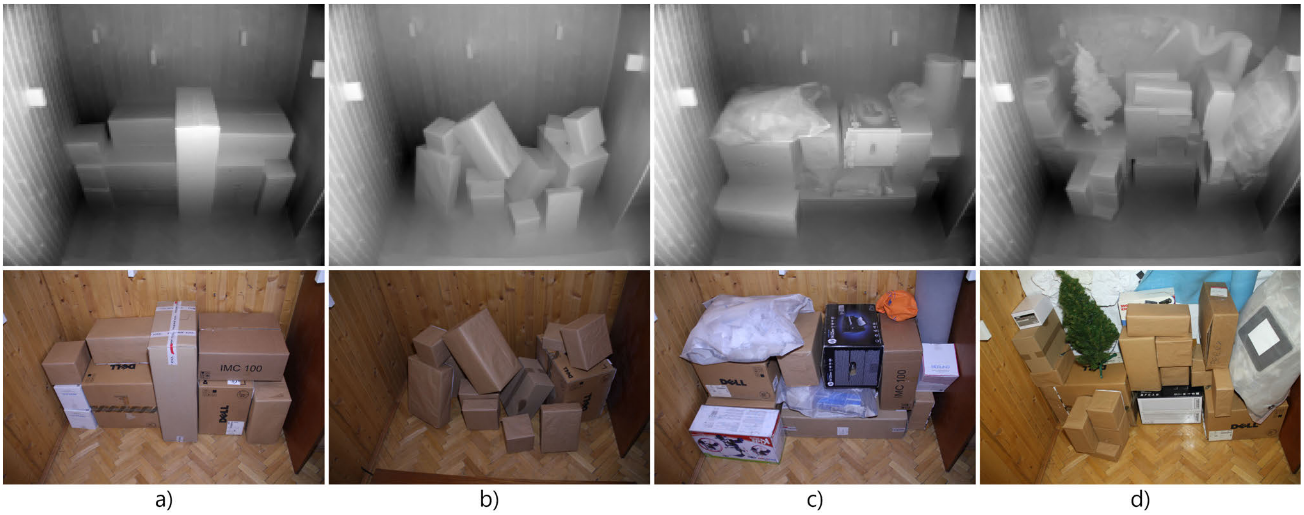

FIGURE 2. Example scenes from the custom dataset. Top: Distance maps of scenes with a normalized value range. Lighter pixels represent points closer to the scanner. Bottom: Color photographs of the corresponding scenes. The dataset contains scenes with packages arranged in a package wall configuration (a), or arbitrary order (b). Besides cardboard packages, the dataset contains cardboard packages with different reflective materials (colorful packages in c); large, semi-transparent shipping bags (c, d); bags of arbitrary shapes and sizes (orange bag in c), and objects of irregular shape (carpet roll, tree in d). The scenes contain one of two types of background: planar (a, b, c), simulating most common trailer interiors; and variable, non-planar background (d) simulating cases where trailer walls are fully occluded by objects of arbitrary shapes.

boxes, shipping bags, and irregular objects (cylindrical packaging, carpets, etc.), with a variable scene background. Distance maps (top) and photographs (bottom) of different scenes are shown in Fig. 2. The packaging items are arranged in one of two configurations: package walls representing ordered packages, and arbitrary order representing cases of tumbled packages that may occur during transportation or unloading errors. The shipping bags are standard, widely-used partially transparent bags, filled with either non-rigid materials that create a large number of edges on the bag surface, or smaller packages that shape the bags into multiple planar segments. The dataset contains two types of background: planar, which represents the most common types of trailer interiors, and non-planar where trailer walls are fully occluded by objects of arbitrary shape.

The dataset consists of distance maps of size 144 pixels × 176 pixels. The pixel values are the distances of the scene points to the scanner (Fig. 2, top). For a clearer understanding of the scene contents, color photographs of the scenes are provided alongside the distance maps (Fig. 2, bottom). The distance maps are generated using a pulse-based ToF depth scanner (device model: SICK Visionary-T DT, type V3S130-2AAAAAA). The scanner generates a distance map (range image) of the scene in a single shot within 20ms. To provide initial noise reduction, for each scene, we acquired 20 consecutive distance maps over 400ms. The final distance map included in the dataset is obtained by averaging the 20 distance maps.

To simulate a real-life loading/unloading scenario, the scene used for creating the dataset (Fig. 2) is designed to resemble the interior of a shipping trailer. The dimensions of the scene filled with packages are 1.65m × 1.95m × 1.2m (height × width × depth), and the average distance of the closest object in the scene to the scanner is 1.2m. To create the dataset, we used 40 packages with different dimensions, along with 5 standard partially transparent shipping bags, and other types of soft bags and objects of irregular shapes. The package sizes vary greatly, from 92cm × 51cm × 28cm to 13cm × 18cm × 25cm, and the aspect ratios of box sides range from 1:1 to 3.3:1. The size of the shipping bags is 85cm × 95cm. Following from the scene dimensions and the scanner resolution, at a distance of 1.2m from the scene the distance between two measured points is 2cm. Depending on the orientation of the box sides (whether they are directly facing the scanner or not), and degree of occlusion, even large package sides can be represented with a small number of points.

The created dataset consists of 272 scenes in total, where 240 of the scenes contain completely different package configurations in arbitrary order. The remaining scenes form two sequences of 16 scenes each, where the first scene shows the space fully loaded with different packaging types, and the next 15 scenes are created by consecutively removing one package from the previous scene. These sequences simulate the unloading process, and enable testing the algorithm's performance in a real-life unloading scenario, where the vision system of an automatic loader/unloader is required to correctly segment the boxes which can be removed at the moment – boxes at the top front of the scene. Removing the top front boxes improves the visibility of occluded boxes in the next scene, and successful segmentation of the scene would mean all boxes were correctly detected by the time the scene is empty.

2) DISTANCE MEASUREMENT ERRORS

Distance measurement errors of ±3cm are expected for objects at a distance of up to 3m, as stated by the scanner manufacturer (device model: SICK Visionary-T DT). Since the measurement errors and distortions are emphasized by the large number of reflection points present in the tight enclosed space of a transport trailer, we observed errors ranging from 5 – 15cm during the creation of the dataset. Besides the general distance measurement errors stated by the manufacturer, four additional types characteristic of ToF-based scanning are observed in the dataset: rounding of inner edges, displacement of outer edges, displacement of whole box sides along the depth axis, and irregularly erroneous measurements of highly reflective surfaces, such as plastic tape and labels. The types of errors observed are shown in Fig. 3 and Fig. 4.

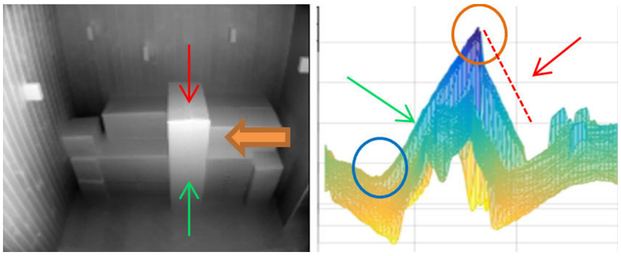

FIGURE 3. ToF-specific distance measurement errors and effects. Left: distance map of a packaging scene. Right: 3D view of the central box where distortions are clearly visible. The orange arrow on the distance map marks the direction of the view of for the 3D representation – the viewpoint for the 3D representation is from the right side of the scene. The front box side is marked with a green, and the top box side with a red arrow. The blue circle marks the rounding of an inner edge, and the orange circle marks an emphasized (sharpened) outer edge. The true position of the top surface in the 3D view is marked with a red dashed line, showing that the top surface is displaced farther away than its true position. Color photograph of the scene is given in Fig. 2(a), bottom.

The rounding of the inner edges spanning over a local environment of up to 5 pixels, and the displacement of the outer edges which can range from 20 – 100mm, are marked in Fig. 3. Fig. 3 also highlights the continuous displacement of whole surfaces, making them appear farther than their actual position. As a result, the intersection of two package sides does not correspond with the true location of the edge. Fig. 4 further demonstrates the extent of measurement errors on reflective surfaces through an example of greatly erroneous measurements on a planar surface made of reflective material (marked in blue). We observed that the orientation of the surface is a significant factor in the magnitude of the measurement error – edges facing the scanner and surfaces at a steep angle to the optical axis of the scanner are affected most. The distortion effects are further emphasized by the large number of reflection points present in the

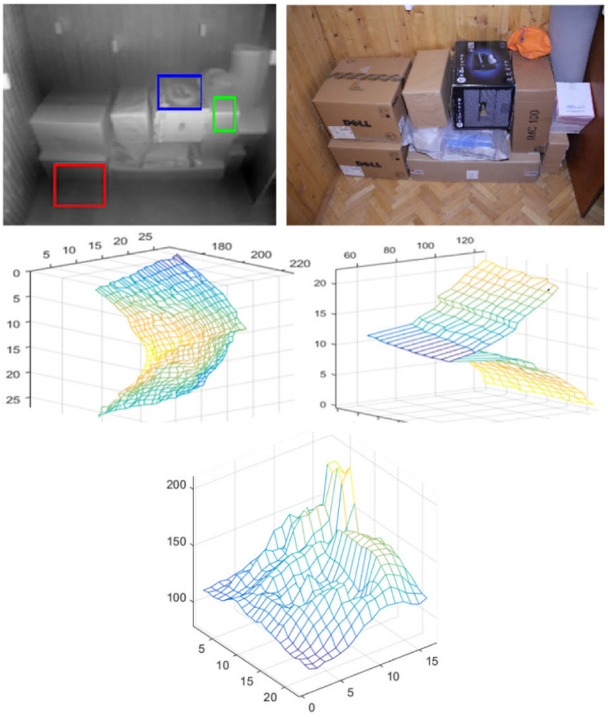

FIGURE 4. Distortion of inner and outer edges and reflective surfaces. Top: distance map with marked areas of interest, and color photograph of the scene. Middle left: 3D view of the area marked with a red rectangle showing rounding of the inner edge. Middle right: 3D view of the area marked with a green rectangle showing emphasized outer edge, which in the real world is formed by perpendicular surfaces. Bottom: 3D view of the area marked with a blue rectangle showing that a planar package side made of reflective material has a non-planar representation in the distance map.

small, enclosed space of a transport trailer filled with packages.

The emphasis of outer edges facilitates separating adjacent surfaces. However, the rounding of inner edges poses a significant problem in separating the surfaces sharing the edge, since the change in distance levels in a small local environment is unnoticeable. As a result, the traditional edge detection methods are unable to detect the inner edges in the distance maps. Furthermore, our previous experiments have shown that the significant irregularities in the measurement of reflective surfaces make it impossible to successfully segment complete package sides with plane segmentation methods based on DSP techniques (RANSAC [42], wavelets [43], etc.).

In this paper we aim to utilize the data from readily available pulse-based scanners, and create a unified algorithm for package segmentation that will successfully adapt to all types of distortion errors, eliminating the need for heavy preprocessing and correction. To that end, we propose a CNN-based algorithm for initial segmentation of package parts in the presence of surface distortion. The annotation process and ground truth labels are described in the sections detailing the CNN-based parts of the algorithm – Section IV-B for segmentation of package sides, and Section IV-C for detection of package edges.

B. SEGMENTATION OF PACKAGE SIDES

1) GENERATING GROUND TRUTH DATA

The ground truth data consist of binary masks marking all box sides in the scene (fully visible and partially occluded boxes of varying sizes and orientations). Each connected component in the binary mask corresponds to a box side. The ground truth data for the distance map shown in Fig. 2(a) is given in Fig. 5, where the annotated box sides are marked in yellow. As seen in Fig. 5, the surface masks (marked in yellow) do not extend to the edges of the box sides. Instead, they cover only the cores of the box sides. The borders of the ground truth surfaces are at a distance of 2 pixels from the edges of the box sides, and any resulting ground truth surface annotated in this manner that is too small (has height or width below 2 pixels, or surface area below 5 pixels), is not included in the ground truth mask. Ground truth data are provided for 253 scans (4586 box sides). The box sides are account for 8% of the total number of pixels in the dataset.

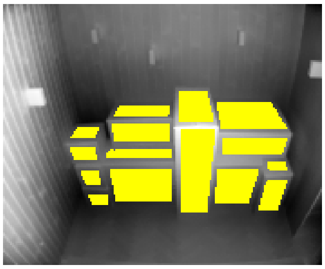

FIGURE 5. Ground truth data for the distance map in Fig. 2(a). The annotated box side cores are marked in yellow.

2) DATA PREPARATION

The barrel distortion is noticeable only on large planar surfaces (trailer floor, walls). Therefore, perspective correction is not applied to the distance maps, as it would have little effect on the results. Previous works have shown that CNNs can successfully model complex perspective transformations [63], which implies that the CNN will be able to adapt to distortions in the depth representation, while successfully performing its primary objective.

Raw distance maps normalized in the range of 0 – 1 are used as input to the CNN. Cropping the distance maps (method first introduced in [51] as overlap-tile in order to create a model usable for different image resolutions) acts as an augmentation method, as it introduces more variety in the input samples by showing different parts of the scene. The distance maps are cropped into 4 patches of 94 rows and 110 columns each (marked with colored borders in Fig. 6), forming 4 input samples.

The training set consists of 140 randomly chosen scans. The validation set consists of 33 scans, and the testing set of 80 scans. The goal of creating a CNN that can be successfully trained on a small dataset with high variability introduces limitations in both the process of data preparation and the complexity of the CNN architecture. Several techniques

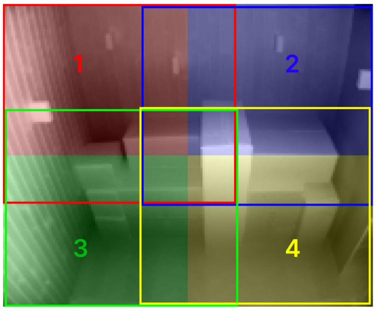

FIGURE 6. Forming 4 training samples from a distance map. The colored borders mark the 4 partially overlapping patches that are used as input to the CNN. Each input patch provides a quarter of the output probability map for a scene (marked by the overlay of the corresponding color).

were employed to overcome overfitting issues. Firstly, a data augmentation strategy of applying different random combinations of transformations (rotation, shifting, zoom, shear, and horizontal flip) to each batch of training samples during every epoch maximizes the number of different data transformations, effectively simulating a much larger dataset. Introducing variation in the training data proved crucial for successfully training a network with the small dataset. Additionally, the CNN design is constrained to a simple design and a small number of parameters, and is thoroughly tested on a dataset of sufficient size (31% of the total data, including various object types, sizes, and positions). L2 regularization is employed during training.

3) PROPOSED CNN STRUCTURE

The primary goal of the CNN is detection of box sides through an optimal compromise between good box side surface coverage, and minimal leakage onto surrounding surfaces. Due to the small number of original images in the dataset (without data augmentation), the main requirement for successful training without overfitting is a low number of parameters.

By segmenting only surface cores, we are able to use a semantic segmentation model for solving the instance segmentation problem of segmenting separate box sides. Segmenting surface cores results in box side masks with clearly distinguishable borders between surfaces, and each connected component can be declared a box side. Since the surface orientation is crucial in determining which box sides are parts of the same package, the CNN outputs surface cores, thereby excluding the heavily distorted areas near the edges that may introduce errors in computing the surface orientation. Furthermore, the simultaneous segmentation of all box sides in a single forward pass of the network gives this algorithm a significant time advantage compared to iterative methods.

The diagram in Fig. 7 shows the structure of the proposed box side segmentation CNN. The encoder (contraction path) consists of 4 convolution blocks, marked in yellow. The downsampling layer in each block reduces the feature map dimensions by 2 with a non-overlapping window. The encoder is followed by two convolutional layers, marked in orange. The decoder (expansion path) consists of 2 convolution blocks marked in green. The upsampling layer in each block increases the feature map dimensions by 4, to generate class labels for each pixel in the input image. The convolution and transposed convolution layers operate on zero-padded input feature maps with filter size 3 × 3, and are followed by ReLU activations. The output convolutional layer with a single filter followed by sigmoid activation produces a probability map of box sides for each input distance map. The probability map for the distance map in Fig. 2(a) generated by the proposed CNN is shown in Fig. 8.

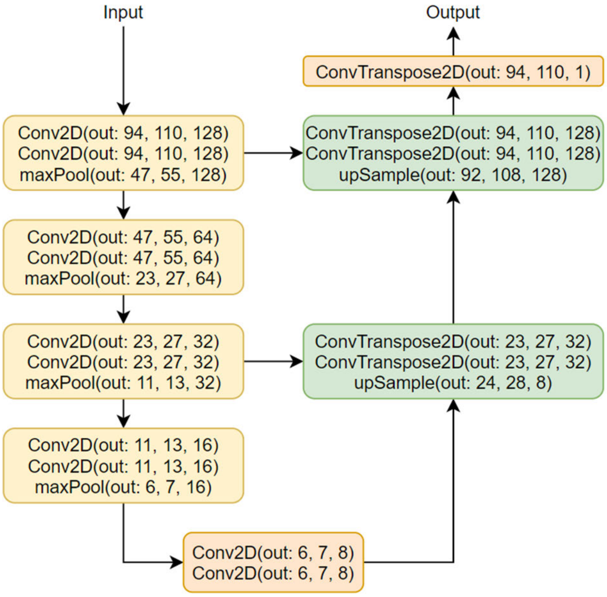

FIGURE 7. Proposed CNN structure for segmentation of box sides. The first two numbers from the output dimensions represent the height and width of the feature maps generated by the layer; and the third number represents the number of convolution filters in the layer.

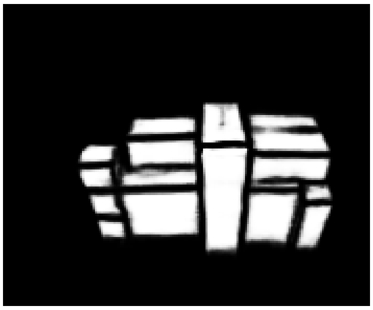

FIGURE 8. Probability map of box sides in the distance map in Fig. 2(a), generated by the proposed CNN given in Fig. 7.

The main motivation behind the asymmetric CNN structure is reducing the number of parameters, thus minimizing execution time. Using 4 pooling layers to greatly reduce the spatial feature map dimensions increases the receptive field of the CNN, and thereby improves the CNN's ability to separate large surfaces sharing an inner edge, since the change in gray levels around inner edges is imperceptible in a small local environment. Due to the significant loss of information by pooling, the two horizontal skip connections are crucial for retaining features of the small objects in the scene. Removing the skip connection before the first pooling layer significantly reduced the precision of segmenting edge pixels and small surfaces.

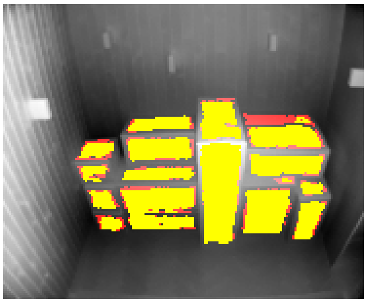

FIGURE 9. Segmented box sides in the distance map in Fig. 2(a). The CNN output is marked in yellow, and the area added to the surfaces after surface expansion is marked in red.

The resulting probability maps (Fig. 8) are binarized with a fixed threshold of 0.5 (Fig. 9, marked in yellow). Surfaces consisting of fewer than 5 pixels are removed from the final output mask.

4) OPTIMIZATION DETAILS

The CNN was trained for 100 epochs with the ADAM (ADAptive Moment estimation) optimization algorithm [64], with a learning rate of 10^{-4} and binary cross-entropy cost function. A mini-batch of 2 samples is chosen to enable the model to retain rare, but crucial features observed only in a small number of samples. Balancing the class weights according to the class frequency did not provide a significant change in results.

5) PROBLEM-SPECIFIC PERFORMANCE METRIC

In the context of box side segmentation, the importance of correctly classifying a pixel depends on the pixel's position (the distance to the nearest surface border). The background pixels separating two surfaces are crucial. Therefore, the weight of the pixels that do not belong to a surface depends on the distance to the two nearest surfaces. The distance to the two nearest surfaces is calculated as the sum of the distances to each surface. Based on these rules, a pixel weight map (Fig. 10) with values normalized in the range of 0 – 1 is calculated for each ground truth mask, and used for evaluating the model performance.

6) SURFACE EXPANSION

In order to provide connected components for each box side, the CNN is trained to output box side cores avoiding pixels near the surface borders. This results in eroded box sides occupying only the surface core (marked in yellow in Fig. 9). We designed a region growing algorithm to perform surface expansion, thus enabling more accurate plane fitting to determine the orientation of the box sides, while still rejecting the edge areas distorted by depth scanning errors. The surface expansion algorithm iteratively checks all neighboring pixels of the particular surface against two criteria: 1) distance of the pixel from the plane fitted in the initial surface, and 2) angle between the normal vectors of the initial CNN-generated surface and the plane fitted in the local 3 × 3 neighborhood of the pixel. The pixels added to the CNN-generated surface cores after expansion are shown in Fig. 9, marked in red. Merging of sides is not allowed in this step.

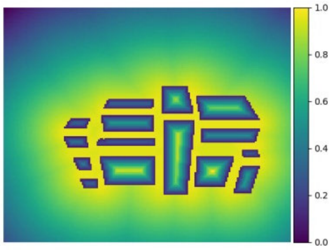

FIGURE 10. Pixel weight map for the annotated box sides shown in Fig. 5, used for calculating the custom problem-specific performance metric.

C. DETECTION OF PACKAGE EDGES

The box side segmentation CNN (Section IV-B) provides masks containing only the surface cores, and cases of surfaces merged in areas of short, rounded inner edges can be detected in the output. To obtain precise localization of the package borders and more accurate segmentation in the areas around rounded inner edges, we perform edge detection in the distance maps using a CNN-based edge detection algorithm that consists of a CNN optimized for training with a limited number of samples, followed by a custom edge thinning algorithm. The edge detection algorithm is described in our previous work [62]. Since the inner and outer edges are marked by significantly different characteristics (as discussed in detail in Section IV-A), the CNN performs classification of inner and outer package edges into two separate classes, thus reducing the intra-class variance, and simplifying the problem of detecting rounded inner edges. As a result, the edge detection CNN produces complete and precise detection of all edges, providing superior accuracy to the box side detection CNN in the areas of weak edges and heavily rounded inner edges. In the following Section IV-D, the results of the box side segmentation CNN and the edge detection CNN are combined to obtain refined box side masks.

D. SURFACE MASK REFINEMENT

Distance measurement errors and heavy surface distortion may result in several types of errors observed in the results of the surface and edge segmentation CNNs – mainly merging, and oversegmentation of box sides. Although rare (observed in fewer than 3% of the test samples), these types of errors are critical since they result in erroneous box forming (merging of boxes or significant errors in estimating the box size), and eliminating them would improve the outcome of crucial situations. Two post-processing algorithms based on combining the results of the segmentation CNNs (surfaces and edges) and DSP techniques, described in detail in the following Sections IV-D1 and IV-D2, are introduced to eliminate the errors.

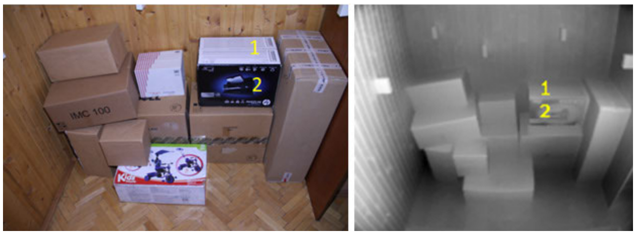

FIGURE 11. Photograph and distance map of a scene containing a package made of highly reflective material. The visible sides of the box are marked with 1 and 2. The edge between the sides is non-distinctive in the distance map.

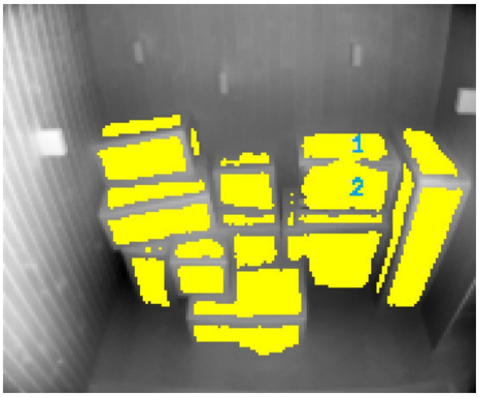

FIGURE 12. The output of the box side detection CNN (marked in yellow) for the scene in Fig. 11. The masks of the two large box sides that belong to the same box (marked with 1 and 2) are partially merged in an area of heavy outer edge distortion.

1) EDGE-BASED BLEEDING AND MERGING ELIMINATION

Several cases of small or severely distorted surfaces merged with neighboring surfaces, and surfaces leaking onto other types of packaging, can be observed in the results of the box side segmentation CNN. Fig. 11 shows the photograph and distance map of a scene containing a package made of highly reflective material and contrasting colors, whose visible sides are marked with 1 and 2. This causes the edge formed by the two sides to be non-distinctive in the distance map, as seen in Fig. 11. Fig. 12 shows the output of the box side detection CNN for this scene, marked in yellow. The surface masks for the two reflective box sides are partially merged. The edge detection CNN correctly detects the complete edge formed by the two sides (Fig. 13). Thus, separating areas in the surface mask along the detected edges eliminates cases of merged box sides.

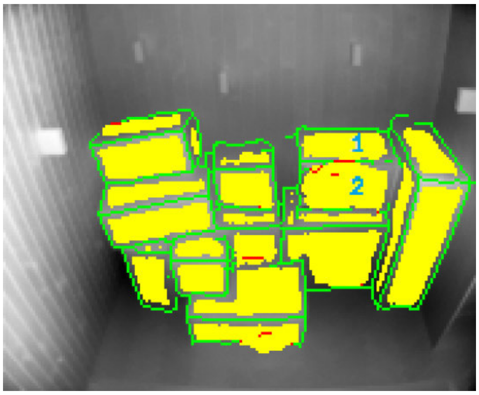

FIGURE 13. The two merged sides marked with 1 and 2 (Fig. 12) are correctly split by the edge detection CNN (along the red line). The pixels representing box sides are marked in yellow, the edge pixels in green, and the pixels where the edges overlap with the surfaces (lines where surfaces should be split) are marked in red.

2) BOX SIDE MERGING

The box side segmentation CNN can result in surface oversegmentation as a result of distance measurement errors caused by reflective sticky tape and labels, or images and text printed on the boxes. The box side merging algorithm is based on two criteria:

- surface adjacency (the surfaces should be adjacent to be part of the same box side)
- surface orientation (the normal vectors of the surfaces should be parallel)

Both criteria rely on empirically determined threshold ranges selected to compensate for the perspective distortion and distance measurement errors. Merging of surfaces separated by edges generated from the edge detection CNN is not allowed. The box side merging algorithm corrects nearly all cases of oversegmentation.

V. FORMING PACKAGES

The algorithm described in this section combines the box sides detected in the previous steps into boxes to provide the final result – position and orientation of all packages in the scene. A box consists of at most three sides perpendicular to each other, where each side forms an outer edge with the other sides. Therefore, the crucial properties to consider when determining if the detected box sides belong to the same box are the angle between the detected box sides, adjacency of the sides, and the type of edge the sides are forming.

A. DETECTING BOXES WITH ONE VISIBLE SIDE

The simplest form of a box in the depth map is a box consisting of only one visible side directly facing the scanner. Since this box side cannot satisfy the box forming conditions with any of the other detected sides, detecting sides facing the scanner reduces the processing time by eliminating a large number of checks of box forming criteria.

The ray vector of a surface represents the vector with an initial point at the center of gravity of the surface and a terminal point at the origin of the coordinate system (location of the scanner). In a distance map, the normal vector of a surface facing the scanner would also be pointing directly towards the scanner. Therefore, a box side facing the scanner is detected by checking whether the angle between the normal vector and ray vector of the surface is smaller than a predefined tolerance threshold. The tolerance threshold is introduced to compensate for distance errors and distortion. Fig. 14 shows a comparison of the directions of normal vectors and ray vectors of two surfaces – one facing the scanner (marked with 1), and one facing the top of the scene (marked with 2). Boxes with only one visible side which is not facing the scanner may appear due to occlusion of the other sides by other objects. Since there is no direct way to detect such box sides, they have to be checked against all box forming criteria with the neighboring sides.

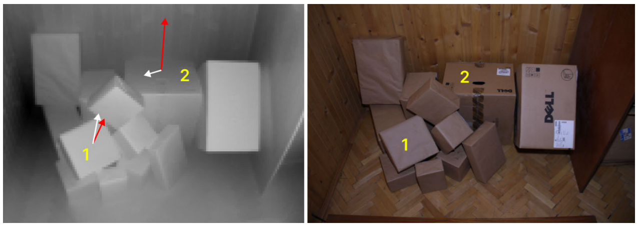

FIGURE 14. Left: Directions of the normal vectors of the surfaces (marked in red) and ray vectors (marked in white) in a distance representation of a scene. Lighter pixels represent points closer to the scanner. The angle between both vectors of the surface facing the scanner (surface 1) is small, whereas the vectors of the surface that does not face the scanner (surface 2) have completely different directions. Note that the viewpoint for this image is moved away from the location of the scanner to enable better visibility of the vectors (if the viewpoint is at the same location as the scanner, the ray vectors would be represented as points). Right: Color photograph of the scene. The corresponding box sides are marked with the same number.

B. FORMING BOXES WITH MULTIPLE VISIBLE SIDES

The number of visible sides of a box depends on the viewing angle of the scanner. Therefore, the next step of the algorithm is finding the sides belonging to the same box. Five main criteria have to be fulfilled in order to declare that two box sides are a part of the same box:

1. The surfaces have to be neighboring – each box side can form a box only with the neighboring box sides. Determining surface adjacency in 3D is challenging in the presence of displacement of whole surfaces along the depth axis. Since the displacement doesn't affect the adjacency of the surfaces in the distance maps, we perform the adjacency check in the distance maps. Two surfaces at a maximum distance of 5 pixels in the distance map are declared neighboring (adjacent).
2. The normal vectors of the surfaces have to be orthogonal, with a given range of tolerance to compensate for surface distortion.

To form a closed cuboid shape, each pair of box sides has to form an outer edge. An outer edge is marked by the fulfillment of the following two criteria (3 and 4):

3. Selecting either one of the two surfaces currently tested as the starting (first) plane, we calculate the projections of all points of the second plane onto the first plane. All points of the second plane (with a given tolerance for a small number of outliers) have to be farther away from the origin (scanner location) than their corresponding projections onto the first plane.
4. The center of gravity of each box side has to be at a greater distance from the origin compared to its orthogonal projection onto the edge the two box sides form.

Criteria 1 – 4 allow for surfaces having neighboring parts but not necessarily sharing an edge, such as two orthogonal surfaces with neighboring corners. Criterion 5 is introduced to eliminate these cases:

5. All points from both box sides are projected onto the intersection line of the two planes fitted to the box sides. The spans of the projections of the box sides onto the intersection line have to overlap at least a given percentage. The overlap percentage threshold is decided based on a balance between including cases of mild surface distortion and one partially covered side, and preventing perpendicular sides of adjacent boxes to be mistakenly classified as a single box.

The criteria have to be fulfilled for all pairs of sides forming a box.

Surface distortion and scanning errors may cause several types of errors in forming boxes considering small, distorted surfaces. Surfaces represented with a small number of points (small surfaces, or surfaces positioned at a small angle to the optical axis of the scanner) can be heavily distorted, causing the surface normals to deviate from a 90° angle, in which case they are regarded as separate boxes. Surface displacement along the depth axis may lead to errors in detecting the position of the shared edge (intersection of the two box sides), which is crucial for criteria 3-5. Highly reflective surfaces with irregularly erroneous distance measurements may cause problems with all of the specified criteria.

C. FORMING OBJECT BORDERS

After the initial segmentation (Section IV), the box side surfaces cover only the surface cores and do not reach the box side edges, and the edges are not required to form closed contours (Fig. 15). Complete detection of the box sides and precise detection of the surface borders is necessary to correctly determine the object position, volume, and orientation. A box side expansion algorithm forms complete, closed borders around every box side surface by adding an edge pixel where the criteria for side expansion are not met. The final result of the box segmentation algorithm for the distance map in Fig. 2(a) is shown in Fig. 16. The complete box side edges are used to determine the absolute box position, volume, and orientation, by detecting and projecting two opposite corners of the box to the world coordinate system.

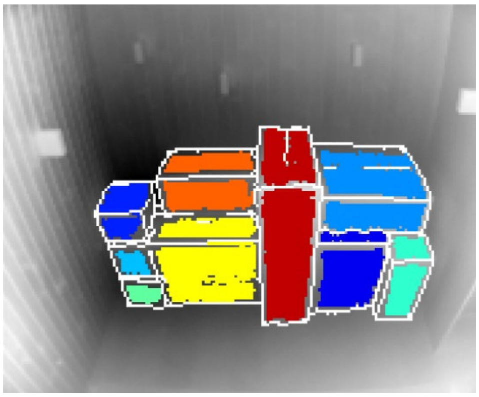

FIGURE 15. Segmented boxes in the scene in Fig. 2(a). Thinned box edges are marked in white, and surfaces representing sides of the same box are marked with the same color. The surfaces do not reach the edges, and the edges do not form closed contours around each box side.

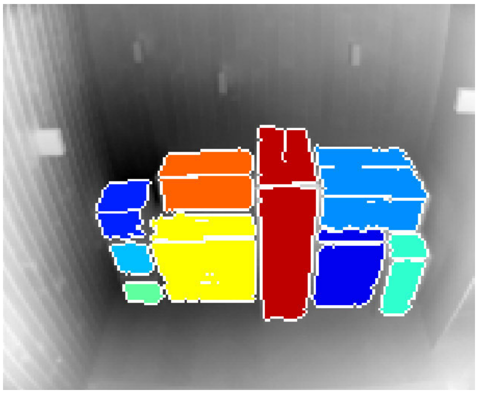

FIGURE 16. Final result of box segmentation with fully formed object borders from the scene in Fig. 2(a). Box sides are complete and fully bounded by edges. Box edges are marked in white, and surfaces representing sides of the same box are marked with the same color.

D. IMPLEMENTATION DETAILS

The training and testing of the proposed algorithm was performed on a computer with the following specifications: Intel Core i7-8700K CPU, 32GB RAM, NVIDIA 1080Ti GPU. The time for training the package surface segmentation CNN (Section IV-B3) for 100 epochs was 5 minutes. The time for training the package edge segmentation CNN (Section IV-C) for 350 epochs was 12 minutes. Inference time per input depth map of the proposed package side detection CNN, including prediction for each image patch and stitching of the patches, is 6ms. Inference time per input depth map of the edge detection CNN is 4ms. The execution time for the DSP-based box forming algorithm is significantly lower. The package detection algorithm is implemented using Python with Tensorflow-Keras and OpenCV.

VI. RESULTS AND DISCUSSION

A. SEGMENTATION OF PACKAGE SIDES

The performance of the proposed algorithm is calculated on a test set of 80 distance maps containing packages, bags and irregular objects, organized in different configurations. The segmentation accuracy of package sides (percentage of correctly classified pixels) is 97.8%, and the box side detection accuracy (percentage of detected box sides) not taking into account if they are correctly combined into boxes is 93.8%. All observed errors are a result of surfaces too small in size, or extreme surface distortion.

The CNN successfully separates object instances, despite lacking the standard structure of a complex instance segmentation network. In the context of package unloading, it is crucial to correctly segment the boxes in the top front of the scene which are the first to be taken out. Visual inspection of the results proves that box sides of all top front boxes represented by a sufficient number of distance points are correctly segmented, with a high difference in probability between the positive and negative classes (as observed in Fig. 8). The proposed CNN (Section IV-B3) outperforms the other CNNs in separating adjacent surfaces sharing an inner edge. The CNN performs correct classification of pixels belonging to partially transparent shipping bags as negative, regardless of the bag contents (soft filling with non-rigid shape, and small rigid boxes). The box sides are successfully distinguished from other planar surfaces, such as planar trailer interiors, and planar surface parts of bags filled with boxes. The CNN successfully adapts to various depth measurement errors on reflective surfaces, and the box side expansion further improves the results by providing better surface coverage, enabling more precise localization of the detected box sides.

Several cases of split surfaces and incomplete surface coverage are observed in the test set. However, these cases have no significant effect on the estimation of the box sides' size and position (Fig. 17c). The cases of split surfaces occur only as a result of large distance measurement errors on reflective surfaces (e.g. reflective tape). Separating two adjacent planar surfaces is not possible using only low-resolution distance maps, and requires additional information (e.g. photographs).

The performance of the proposed CNN (Section IV-B3) is compared to three U-Net-based models: the original U-Net architecture; a U-Net architecture with a reduced number of filters (maximum of 128 – in the last layer); and proposed inverse model. The inverse model shares the proposed architecture, with the difference of doubling the number of filters after each convolution block (maximum of 128) instead of halving. The architecture specifications (number of filters per layer) of the reduced U-Net and proposed inverse architecture were selected to achieve a total number of parameters as similar to the proposed CNN as possible. All CNNs are trained on our custom dataset with the same training-validation-test split. The CNNs are trained using ADAM [64], binary cross-entropy cost function, and a mini-batch of 2 samples. An optimal learning rate for each CNN was selected through grid search.

We compare the performance of the proposed CNN to the original U-Net architecture, which has demonstrated top results in segmentation of single-channel images, and effectiveness with small training datasets. With the reduced

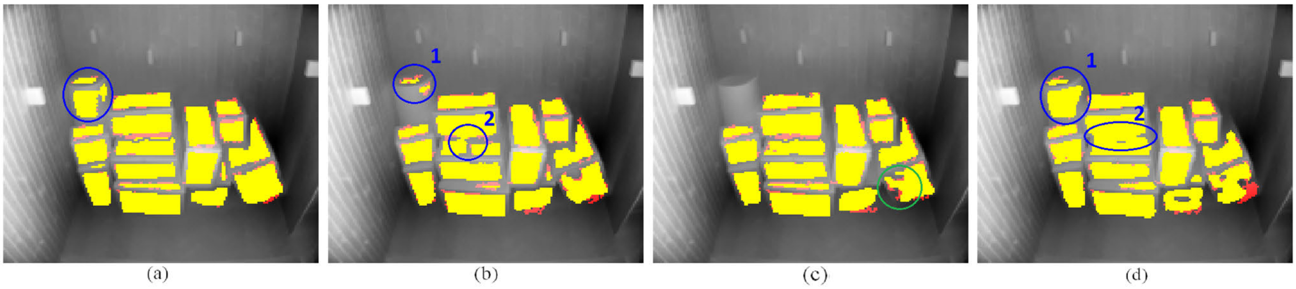

FIGURE 17. Results of surface segmentation with the CNN specified in Section IV-B3: a) U-Net, b) Reduced U-Net, c) Proposed, d) Proposed inverse. The CNN output is marked in yellow, and the area added to the surfaces after surface expansion is marked in red. Although the U-Net results seem most accurate at first glance, the segment circled in blue shows a crucial mistake of detecting a cylindrical object as two planar segments. This error is absent only in the result of the proposed CNN (c). In b), circle 2 marks surface oversegmentation, and merging of two large surfaces along an inner edge. The surface marked in green in c) is eroded due to depth errors on tapes. However, this enables a more precise calculation of the surface normal. In d), circle 2 marks complete merging of surfaces along an inner edge. A photograph of the scene is shown in Fig. 18.

U-Net architecture we aim to reduce the chance of overfitting and provide a fair comparison with the proposed model. With the proposed inverse architecture, we aim to show the benefits of having a larger number of filters in the bottom layers for datasets with high variability. Having a similar number of parameters enables comparison of the architecture changes independent of computational capacity, thus ensuring that the observed performance differences result strictly from the architecture changes, rather than the overall change in processing power due to the number of parameters.

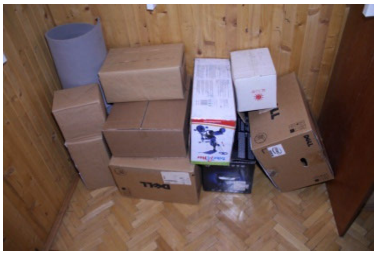

FIGURE 18. Color photograph of the scene analyzed in Fig. 17.

Fig. 17 shows the results of package side segmentation on the scene shown in the color photograph in Fig. 18. As observed in Fig. 17, the proposed CNN performs well in crucial tasks, such as successfully separating surfaces sharing a gradual, rounded inner edge, and rejecting irregular objects. Furthermore, the proposed CNN shows a significantly lower sensitivity to small imperfections of the planar surfaces.

Table 1 quantifies the performance of the proposed method in comparison to previous image segmentation algorithms and experiments based on the U-Net architecture. We evaluated the performance of the algorithms using four different metrics:

- Accuracy (ratio of the number of correct predictions to the total number of input samples)
- Pixel-wise F1 Score (harmonic mean of precision and recall)
- Average Precision (weighted mean of precisions calculated at 11 different equally-spaced recall levels)
- Weighted accuracy (accuracy with different importance assigned to pixels based on their location, described in detail in Section IV-B5)

TABLE 1. Performance comparison of the proposed method and experiments.

CNN	Acc. (%)	Pixel-wise F1	AP	weighted acc.	No. of coef.
U-Net	97.49	0.8759	0.9306	0.6362	21976513
Reduced U-Net	97.27	0.8692	0.9462	0.6070	787217
Proposed inverse	96.12	0.8149	0.9072	0.5156	**353809**
Proposed	**97.81**	**0.8921**	**0.9611**	**0.7055**	649969

All metrics are calculated at the pixel level, where each pixel is treated as an independent sample, rather than at the object level (package sides). The predicted class of each pixel is obtained by binarizing the CNN-generated probability maps using a fixed binarization threshold of 0.5. AP (Average Precision) is calculated on the CNN-generated probability maps, and the other metrics are calculated on the binary surface masks. For a detailed comparison of computational complexity and time efficiency, independent of the exact hardware configuration, the rightmost column of Table 1 shows the total number of parameters of each CNN. The performance metrics take into account incomplete surface coverage, and segmentation errors on all types of packaging and scene backgrounds.

According to Table 1, further confirmed by visual inspection of the results, the proposed CNN provides the best overall performance. The accuracy metric is not optimal for datasets with heavy class imbalance (the box sides class is represented with 8% of the pixels). Therefore, the superiority of the proposed model is more emphasized in the other metrics, and is most noticeable in the weighted accuracy metric which takes into account pixel importance in the context of box recognition. The small number of training samples is not enough to train the complex U-Net with a large number of parameters. Having a larger number of filters in the first

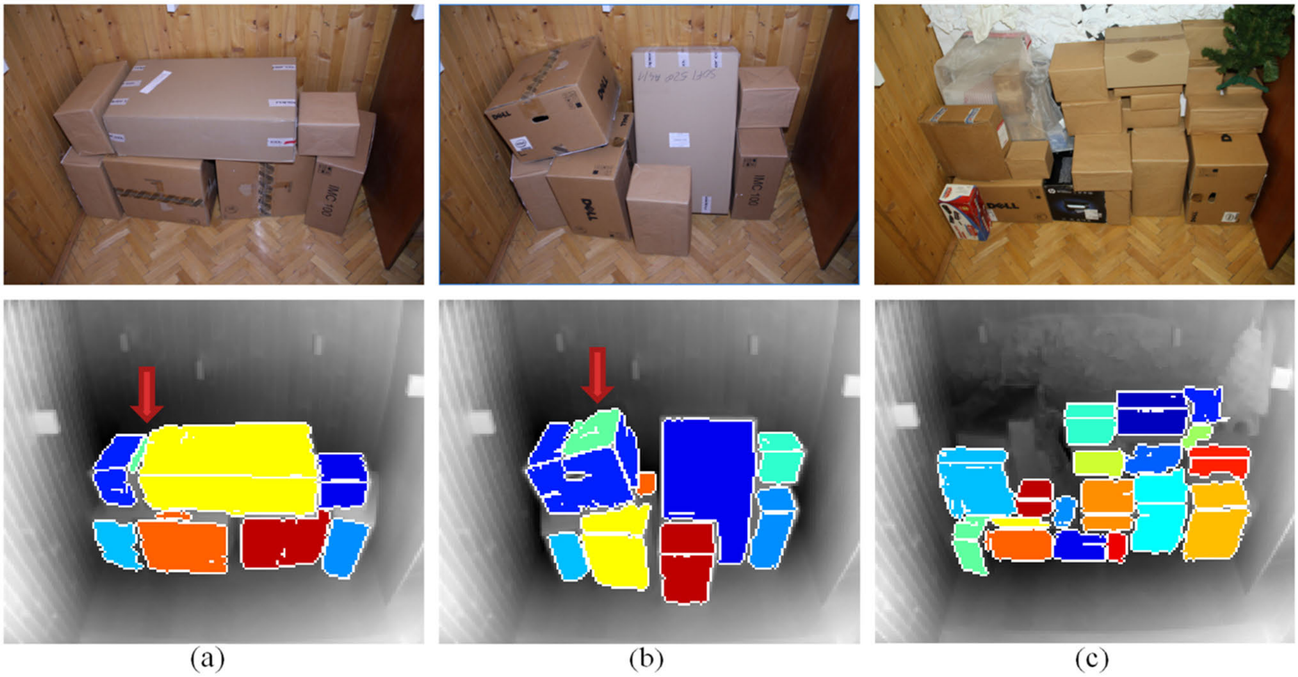

FIGURE 19. Package detection results. Top: color photographs. Bottom: final result of the proposed package detection algorithm for the corresponding scenes. Sides that are a part of the same box are marked with the same color. The package sides represented with a sufficient number of distance points are fully segmented and correctly combined into packages, as shown in a) and b). The unmatched side of the top left package in a) marked in green (pointed at by the red arrow) is small, with high eccentricity, and as such, can be easily removed through a surface eccentricity check. The only unmatched package side in b) is a result of oversegmentation of the top side along the reflective tape. The assumption of at most 3 visible sides prevents the fourth surface (pointed at by the red arrow) to be assigned to the correct package. As seen in c), the proposed algorithm performs well in unorganized piles of packages in scenes with a variable background containing additional packaging items of arbitrary shape, including partially transparent bags. Additionally, free packages are successfully differentiated from the packages inside shipping bags.

layers allows the model to retain a larger number of different low-level characteristics of the dataset which contains high variability due to reflections and distance errors. Decreasing the number of filters in the deeper layers forces the deepest layers to focus only on the features most relevant for defining a box side. This causes the performance increase of the proposed model compared to both the reduced U-Net and Proposed inverse models.

B. SEGMENTATION OF PACKAGE EDGES

The edge detection algorithm is based on an edge detection CNN containing 648190 parameters. The performance of the edge detection algorithm is calculated on a test set of 80 distance maps containing packages, bags and several types of irregular objects, and achieves segmentation accuracy of 97.13%. The algorithm correctly segments box edges represented by a sufficient number of distance points in the presence of heavy distortion in the distance maps. More details on the edge detection algorithm and performance metrics can be seen in [62].

C. FORMING PACKAGES

The performance of the box forming algorithm that combines the package sides into packages is calculated on a test set of 62 distance maps containing packages, bags and irregular objects, organized in different configurations. The box recognition accuracy (percentage of boxes with all visible sides detected and classified into the same box) is 82.5%. Packages with only one visible side are successfully detected, and box sides represented by a sufficient number of distance points are correctly combined into packages. The algorithm successfully handles cluttered environments containing packages inside partially transparent shipping bags, and items of arbitrary shape, as seen in Fig. 19c. Visual inspection of the results shows that packages located at the top front of the scene and represented by a sufficient number of distance points are correctly detected. Removing the top front packages would make small, heavily occluded surfaces at the back of the scene fully visible, allowing them to be correctly detected.

Small surfaces and surfaces with high eccentricity may have distorted normal vectors, which results in those types of surfaces remaining unmatched (Fig. 19a). These cases represent the largest part of the box forming errors. The errors do not affect the accuracy of the calculated package size or optimal point of contact for removal, and can be easily resolved by excluding the small surfaces and surfaces with high eccentricity from the final package mask.

Several cases of falsely matched box sides can be resolved by introducing additional rules in the box forming algorithm based on the geometry and placement of the boxes. For example, the green box side marked with an arrow in

Fig. 19(a) cannot be a separate box, since a whole box cannot fit into the space between the dark blue and yellow boxes. Similarly, the green box side (marked with an arrow) in Fig. 19(b) can only be a part of the top leftmost box, and allowing more than 3 segmented components to form a box would enable matching the green side to the dark blue box sides.

VII. CONCLUSION

The proposed hybrid algorithm for detecting packages in 2D distance maps has proved successful in detecting packages in a cluttered environment filled with packaging objects of arbitrary shape, and partially transparent shipping bags. Structuring the algorithm as an integration of deep learning-based initial segmentation of box sides and edges, and combining the segmented sides to form complete boxes based on the cuboid shape of packaging boxes, enable fast and correct package segmentation.

The two proposed CNN configurations designed for fast initial segmentation of package components, trained with a limited number of samples with heavily imbalanced classes, successfully generalize over the samples with high variability, heavy surface distortion, and distance measurement errors characteristic of ToF-based depth scanning. The CNNs perform correct segmentation of box sides and edges represented with a sufficient number of points, regardless of their size, position, and orientation. The results demonstrate that a carefully designed simple network architecture with a limited number of parameters, combined with data augmentation strategies and regularization techniques, have resulted in CNNs able to generalize over a wide selection of packaging items in different lighting conditions, and successfully overcome overfitting issues due to the limited number of training samples. The CNNs prove superior to the DSP techniques in segmenting package components due to their higher robustness to the depth measurement errors. Furthermore, the CNNs' ability to adapt to perspective distortion minimizes the need for pre-processing.

Forming the ground truth masks such as to contain only the surface cores enables the successful use of a semantic segmentation CNN structure for instance segmentation of planar surfaces. The surface segmentation CNN successfully differentiates the planar surfaces of boxes from planar surfaces of other types of packaging. Formulating the objective of the edge detection as two separate targets represented by largely different features (inner and outer edges) had a great influence in obtaining favorable results. As shown by both the visual results and calculated metrics, simple CNNs with a limited number of training parameters and a decreasing number of filters in the top layers prove best in retaining crucial features in a highly variable small dataset. Alongside precise detection of planar segments in distance maps with highly variable distance measurement errors, the proposed work serves as an initial step for general detection of edges and planar surfaces in distance maps.

The concise set of geometry-based rules for combining the segmented box sides to form packages produce fast and accurate segmentation results regardless of package position, orientation, and partial occlusion. The initial step of detecting boxes with only one visible side results in a significant decrease of the processing time. The final box forming results show that segmentation and detection errors occur primarily in two cases: surfaces too small in size, and surfaces with significantly distorted depth representation due to distance measurement errors. Therefore, the main limitations of the proposed approach stem from the scanning technology and dataset size – the minimum object size is limited by the scanning resolution, and the small dataset size does not allow for complex learning-based solutions for the box forming step of the algorithm. Surfaces represented with a small number of points due to the angle they are positioned at cannot be segmented with a high precision, and as a result, they are discarded, and the package detection decision is made relying on the remaining visible box sides.

Since the errors are concentrated on small and reflective surfaces, simulated data lacking ToF-specific distance measurement errors are rendered unusable for this application. This implies that using more precise scanning technology would result in an additional increase in algorithm performance. Several cases of errors in box forming can be eliminated with additional geometry-based constraints. A promising research direction to address the problem of incorrect distance measurements of reflective surfaces is increasing the training dataset. A significant increase of training data available would provide a possibility for wider use of learning-based methods, thus introducing new possibilities for further improvement of the box forming accuracy of small, heavily occluded boxes and boxes made of highly reflective materials. Overall, the proposed algorithm is a precise, robust solution for package detection, which serves as a solid basis for fully automated loading and unloading of transport trailers.

ACKNOWLEDGMENT

Information and techniques disclosed in this paper are owned by Fast Global Solutions, Inc.

REFERENCES

[1] Boston Dynamics. *Handle*. Accessed: Feb. 6, 2021. [Online]. Available: https://www.bostondynamics.com/handle/

[2] Magazino. *Toru—Autonomes Kommissionieren Kleiner Kartons*. Accessed: Oct. 11, 2022. [Online]. Available: https://www.magazino.eu/produkte/toru/

[3] Copal C2—Copal Handling Systems. *Fast and Efficient Unloading of Goods*. Accessed: Oct. 11, 2022. [Online]. Available: https://www.copalhandlingsystems.com/en/products/copal-c2/

[4] Boston Dynamics. *Stretch—Autonomous Case Handling Robot*. Accessed: Feb. 11, 2022. [Online]. Available: https://www.bostondynamics.com/products/stretch/

[5] RobotWorx. *Motoman MH80*. Accessed: Feb. 11, 2022. [Online]. Available: https://www.robots.com/robots/motoman-mh80/

[6] D. Katsoulas, C. Bastidas, and D. Kosmopoulos, "Superquadric segmentation in range images via fusion of region and boundary information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 781–795, May 2008.

[7] A. Kirchheim, M. Burwinkel, and W. Echelmeyer, "Automatic unloading of heavy sacks from containers," in *Proc. IEEE Int. Conf. Autom. Logistics*, Sep. 2008, pp. 946–951.

[8] H. Thamer, H. Kost, D. Weimer, and B. Scholz-Reiter, "A 3D-robot vision system for automatic unloading of containers," in *Proc. IEEE 18th Conf. Emerg. Technol. Factory Autom. (ETFA)*, Cagliari, Italy, Sep. 2013, pp. 1–7.

[9] R. Schnabel, R. Wessel, R. Wahl, and R. Klein, "Shape recognition in 3D point-clouds," in *Proc. 16th Int. Conf. Central Eur. Comput. Graph., Vis. Comput. Vis.*, Pilsen, Czech Republic, Feb. 2008, pp. 65–73.

[10] C. Weber, S. Hahmann, and H. Hagen, "Methods for feature detection in point clouds," in *Proc. Vis. Large Unstructured Data Sets IRTG Workshop*, vol. 19, 2010, pp. 90–99.

[11] V. Kovács and G. Tevesz, "Corner detection and classification of simple objects in low-depth resolution range images," *Periodica Polytechnica Electr. Eng.*, vol. 57, no. 1, pp. 9–17, Jul. 2013.

[12] S. Buck, R. Hanten, K. Bohlmann, and A. Zell, "Multi-sensor payload detection and acquisition for truck-trailer AGVs," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 718–723.

[13] R. Varga and S. Nedevschi, "Vision-based autonomous load handling for automated guided vehicles," in *Proc. IEEE 10th Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, Sep. 2014, pp. 239–244.

[14] F. Weichert, S. Skibinski, J. Stenzel, C. Prasse, A. Kamagaew, B. Rudak, and M. T. Hompel, "Automated detection of euro pallet loads by interpreting PMD camera depth images," *Logistics Res.*, vol. 6, nos. 2–3, pp. 99–118, Jun. 2013.

[15] P. Doliotis, C. D. McMurrough, A. Criswell, M. B. Middleton, and S. T. Rajan, "A 3D perception-based robotic manipulation system for automated truck unloading," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Fort Worth, TX, USA, Aug. 2016, pp. 262–267.

[16] L. Dörr, F. Brandt, M. Pouls, and A. Naumann, "Fully-automated packaging structure recognition in logistics environments," in *Proc. 25th IEEE Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Vienna, Austria, Sep. 2020, pp. 526–533.

[17] Z. Rozsa and T. Sziranyi, "Obstacle prediction for automated guided vehicles based on point clouds measured by a tilted LiDAR sensor," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2708–2720, Aug. 2018.

[18] U. Shafiq, M. Taj, and M. Ali, "More for less: Insights into convolutional nets for 3D point cloud recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China, Sep. 2017, pp. 1607–1611.

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[20] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 10778–10787.

[21] Y.-C. Chiu, C.-Y. Tsai, M.-D. Ruan, G.-Y. Shen, and T.-T. Lee, "Mobilenet-SSDv2: An improved object detection model for embedded systems," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, Kagawa, Japan, Aug. 2020, pp. 1–5.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Computer Vision—ECCV 2016* (Lecture Notes in Computer Science), vol. 9905. Cham, Switzerland: Springer, 2016, pp. 21–37.

[23] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[24] C.-Y. Wang, A. Bochkovskiy, and H.-Y. Mark Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv:2207.02696*.

[25] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 213–229.

[26] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–16.

[27] J. Yang, C. Li, X. Dai, and J. Gao, "Focal modulation networks," in *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, 2022, pp. 4203–4217.

[28] N. Yang, L. von Stumberg, R. Wang, and D. Cremers, "D3VO: Deep depth, deep pose and deep uncertainty for monocular visual odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1278–1289.

[29] P.-Y. Chen, A. H. Liu, Y.-C. Liu, and Y. F. Wang, "Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 2619–2627.

[30] H. Chen, P. Wang, F. Wang, W. Tian, L. Xiong, and H. Li, "EPro-PnP: Generalized end-to-end probabilistic Perspective-n-Points for monocular object pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2771–2780.

[31] K. Park, T. Patten, and M. Vincze, "Pix2Pose: Pixel-wise coordinate regression of objects for 6D pose estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7667–7676.

[32] X. Zhang, Z. Jiang, and H. Zhang, "Real-time 6D pose estimation from a single RGB image," *Image Vis. Comput.*, vol. 89, pp. 1–11, Sep. 2019.

[33] Z. Fan, Z. Song, J. Xu, Z. Wang, K. Wu, H. Liu, and J. He, "Object level depth reconstruction for category level 6D object pose estimation from monocular RGB image," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 13662, 2022, pp. 124–139.

[34] A. Naumann, F. Hertlein, L. Dörr, and K. Furmans, "Parcel3D: Shape reconstruction from single RGB images for applications in transportation logistics," 2023, *arXiv:2304.08994*.

[35] J. Castaño-Amoros, F. Fuentes, and P. Gil, "MOSPPA: Monitoring system for palletised packaging recognition and tracking," *Int. J. Adv. Manuf. Technol.*, vol. 126, nos. 1–2, pp. 179–195, Feb. 2023.

[36] N. Lopac, F. Hržic, I. P. Vuksanovic, and J. Lerga, "Detection of non-stationary GW signals in high noise from Cohen's class of time–frequency representations using deep learning," *IEEE Access*, vol. 10, pp. 2408–2428, 2022.

[37] A. A. Abdelhamid, E. M. El-Kenawy, B. Alotaibi, G. M. Amer, M. Y. Abdelkader, A. Ibrahim, and M. M. Eid, "Robust speech emotion recognition using CNN+LSTM based on stochastic fractal search optimization algorithm," *IEEE Access*, vol. 10, pp. 49265–49284, 2022.

[38] D. P. Yadav, A. Sharma, S. Athithan, A. Bhola, B. Sharma, and I. B. Dhaou, "Hybrid SFNet model for bone fracture detection and classification using ML/DL," *Sensors*, vol. 22, no. 15, p. 5823, Aug. 2022.

[39] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[40] S. Montaha, S. Azam, A. K. M. R. H. Rafid, M. Z. Hasan, A. Karim, and A. Islam, "TimeDistributed-CNN-LSTM: A hybrid approach combining CNN and LSTM to classify brain tumor on 3D MRI scans performing ablation study," *IEEE Access*, vol. 10, pp. 60039–60059, 2022.

[41] M. Sajjad, Z. A. Khan, A. Ullah, T. Hussain, W. Ullah, M. Y. Lee, and S. W. Baik, "A novel CNN-GRU-based hybrid approach for short-term residential load forecasting," *IEEE Access*, vol. 8, pp. 143759–143768, 2020.

[42] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with apphcatlons to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[43] Z. Jakovljevic, R. Puzovic, and M. Pajic, "Recognition of planar segments in point cloud based on wavelet transform," *IEEE Trans. Ind. Informat.*, vol. 11, no. 2, pp. 342–352, Apr. 2015.

[44] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.

[45] N. Silberman, D. Hoiem, P. Kohli, and F. Rob, "Indoor segmentation and support inference from RGBD images," in *Proc. ECCV*, 2012, pp. 746–760.

[46] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science, vol. 8693, 2014, pp. 740–755.

[47] 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.*, 2017, pp. 5105–5114.

[48] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.

[49] 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.

[50] W. Wang, R. Yu, Q. Huang, and U. Neumann, "SGPN: Similarity group proposal network for 3D point cloud instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2569–2578.

[51] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.

[52] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Switzerland: Springer, 2018, pp. 3–11.

[53] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual U-Net," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 749–753, May 2018.

[54] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.

[55] D. Jha, P. H. Smedsrud, M. A. Riegler, D. Johansen, T. D. Lange, P. Halvorsen, and H. D. Johansen, "ResUNet++: An advanced architecture for medical image segmentation," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2019, pp. 225–230.

[56] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1395–1403.

[57] Z. Yu, C. Feng, M.-Y. Liu, and S. Ramalingam, "CASENet: Deep category-aware semantic edge detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1761–1770.

[58] J. J. Lim, C. L. Zitnick, and P. Dollár, "Sketch tokens: A learned mid-level representation for contour and object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3158–3165.

[59] G. Bertasius, J. Shi, and L. Torresani, "DeepEdge: A multi-scale bifurcated deep network for top-down contour detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 4380–4389.

[60] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool, "Convolutional oriented boundaries: From image segmentation to high-level tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 819–833, Apr. 2018.

[61] L. Han, X. Li, and Y. Dong, "Convolutional edge constraint-based U-Net for salient object detection," *IEEE Access*, vol. 7, pp. 48890–48900, 2019.

[62] E. Vasileva, N. Avramovski, and Z. Ivanovski, "Detection of package edges in distance maps," in *Proc. 28th Eur. Signal Process. Conf. (EUSIPCO)*, Amsterdam, The Netherlands, Jan. 2021, pp. 600–604.

[63] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1696–1704.

[64] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

ELENA VASILEVA (Graduate Student Member, IEEE) was born in Skopje, North Macedonia. She received the B.Sc. degree in computer science and information technologies and the M.Sc. degree in digital signal processing from Ss. Cyril and Methodius University in Skopje, North Macedonia, in 2013 and 2017, respectively, where she is currently pursuing the Ph.D. degree in computer vision with the Faculty of Electrical Engineering and Information Technologies.

Her research experience includes research projects in the field of digital image processing and analysis, and deep learning. Her current research interests include digital image processing, image and video analysis, deep learning, and 3D vision.

ZORAN A. IVANOVSKI (Senior Member, IEEE) received the B.S. degree in information technology and automation and the M.Sc. and Ph.D. degrees in electronics from Ss. Cyril and Methodius University in Skopje, Skopje, North Macedonia, in 1990, 2000, and 2006, respectively. In 1991, he joined the Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje, where he was engaged as a Teaching and Research Assistant. In 2003 and 2004, he was a Faculty Research Associate with the Department of Electrical Engineering, Arizona State University, Tempe, AZ, USA, where he was involved in the development of super-resolution algorithms. He is currently a Full Professor with the Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje, where he is also the Head of the Institute of Electronics. His research experience includes numerous research projects in the area of digital image processing, image analysis, and machine vision, funded by industries, including NXP, Texas Instruments, Fast Global Solutions, Alfa-Zet Systems, PathKeeper Surgical, and Mikrosam. He published over 100 research articles. His research interests include digital image and video processing, image analysis and understanding, and computer vision. He served as the Chair for the SP/EMB Chapter of the IEEE Republic of Macedonia Section (2013–2015).