Damage and Tampering Assessment in Transportation Logistics and Warehousing using Deep Learning

Zur Erlangung des akademischen Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN
(Dr.-Ing.)

von der KIT-Fakultät für Maschinenbau
des Karlsruher Instituts für Technologie (KIT)

angenommene

DISSERTATION

von

Alexander Naumann, M.Sc.

Tag der mündlichen Prüfung:	20. September 2024
Hauptreferent:	Prof. Dr.-Ing. Kai Furmans
Korreferent:	Prof. Dr. Masayoshi Tomizuka

Preface

This research was conducted during my time as a scientific researcher at the FZI Research Center for Information Technology (FZI) and the Karlsruhe Institute of Technology (KIT) in Karlsruhe, Germany. I am deeply grateful to all those who have supported me throughout this journey, making this thesis possible.

I would like to express my sincere gratitude to Prof. Furmans for giving me the opportunity to pursue research in this fascinating field under his supervision. His guidance and support, combined with the freedom to explore my own ideas, have been invaluable to my growth as a researcher. Also, I would like to extend my heartfelt thanks to Prof. Tomizuka for hosting me at the Mechanical Systems Control Lab (MSCL) at UC Berkeley and for serving as co-examiner of this dissertation. My time in Berkeley was both enjoyable and productive, and I would like to acknowledge the support of the Karlsruhe House of Young Scientists (KHYS) for facilitating this research visit.

My sincere appreciation goes to my colleagues at FZI and KIT. I have greatly benefited from our diverse team, and I cherish the friendships I have formed. Moreover, I am deeply appreciative of the support and encouragement of my friends outside work. It means a lot to me to have you all in my life.

Finally, my deepest thanks go to my beloved wife and family. Words cannot express my gratitude for your unwavering support, patience, and love. You have been my source of strength and motivation throughout this journey, and I am eternally grateful.

Karlsruhe, September 2024
Alexander Naumann

Kurzfassung

Aufgrund der stetig wachsenden Versandnachfrage für wertvolle Güter in den nationalen und internationalen Lieferketten gewinnt die automatisierte Analyse von Schäden und Manipulationsversuchen während des Transportprozesses stets weiter an Bedeutung. Eine kontinuierliche und koordinierte Überwachung der transportierten Pakete durch den Einsatz moderner Bildverarbeitungsmethoden kann die rechtzeitige und effiziente Erkennung potenzieller Problemfälle erleichtern. Dies wiederum kann entscheidend dazu beitragen, die Unversehrtheit und Integrität der Fracht innerhalb des Transportnetzwerkes zu gewährleisten.

In der vorliegenden Arbeit wird zunächst ein allgemeiner und detaillierter Literaturüberblick über zahlreiche Anwendungsszenarien von Bildverarbeitungsalgorithmen in der Transportlogistik und Lagerhaltung gegeben. Dieser dient der Einordung des Standes der Technik sowie der Identifikation offener Forschungsfragen. Anschließend wird der Fokus auf die Bewertung von Schäden und Manipulationsversuchen für die Zustellung auf der letzten Meile gesetzt. Dieser Anwendungsfall erfordert hochflexible Ansätze mit geringen Hardwareanforderungen, da dem Kurier oder dem Endkunden in der Regel nur einfache portable Geräte wie Smartphones zur Verfügung stehen. Im Gegensatz zu bestehenden Arbeiten, die komplizierte multisensorische Setups und eine a-priori bekannte Umgebung erfordern, basieren alle in dieser Arbeit präsentierten Ansätze auf einem einzigen RGB-Bild als Eingabe.

Die zuverlässige Erkennung und Lokalisierung von Paketen in Bildern ist eine wichtige Grundlage für nachgelagerte Aufgaben, die sich mit der Bewertung von Schäden und Manipulationsversuchen befassen. Daher wird zunächst eine vollautomatische Pipeline zur Erzeugung von Instanzsegmentierungsdatensätzen

vorgestellt, welche das Training von Modulen zur gezielten Paketerkennung ermöglicht. Aufgrund der Einfachheit des Ansatzes können Datensätze leicht erstellt und aktualisiert werden, wodurch Robustheit gegenüber potenzieller Varianz im Erscheinungsbild der Pakete (z.B. durch länderspezifische Unterschiede) erreicht werden kann.

Anschließend wird eine Pipeline zur Erkennung von Manipulationsversuchen, wie beispielweise das Öffnen und Wiederverschließen eines Pakets zur unerlaubten Inhaltsentnahme, vorgestellt. Die neu entwickelte Pipeline prädiziert dabei zunächst die Position der Paketeckpunkte und nutzt diese Information aus, um sichtpunktinvariante Frontalansichten für alle sichtbaren Seitenflächen des vorliegenden Pakets zu errechnen. Unter der Annahme, dass eine Referenztextur vorhanden ist, kann das Problem der Manipulationserkennung dann auf die Erkennung optischer Oberflächenveränderungen je Paketseitenfläche reduziert werden. Solche visuellen Unterschiede zwischen zwei Paketseitenflächen werden erkannt, indem Bildhomogenisierungstechniken angewandt werden und anschließend Bildähnlichkeitsmetriken als Kriterium herangezogen werden.

Abschließend wird das Problem der bildbasierten Schadenserkennung und -bewertung von Paketen behandelt, wobei ausschließlich Deformationsschäden betrachtet werden. Es wird die neue, dedizierte neuronale Netzwerkarchitektur CubeRefine R-CNN vorgestellt, welche die Prädiktion des minimalen umgebenden dreidimensionalen Quaders eines Pakets mit einem iterativen Verfeinerungsansatz zur Anpassung an Verformungen kombiniert. Der Ansatz prädiziert gleichzeitig die aktuelle, potenziell deformierte Form des Pakets und seine ursprüngliche, quadratische Form. Somit ermöglicht er eine detaillierte Schadensbewertung und -quantifizierung durch den direkten Vergleich zweier 3D Dreiecksgitter. Darüber hinaus wird Parcel3D vorgestellt: ein synthetischer Datensatz beschädigter und intakter Pakete mit vollständigen 2D- und 3D-Annotationen.

Die Leistungsfähigkeit aller Ansätze wird anhand realer Datensätze quantitativ und qualitativ bewertet. Zudem wurden die erstellten Datensätze und der Quellcode veröffentlicht, um die Reproduzierbarkeit der vorgestellten Arbeit zu gewährleisten.

Abstract

Due to the steadily increasing amount of valuable goods in national and international supply chains, automated damage and tampering assessment during the transportation process is continuously gaining importance. Steady and streamlined monitoring of the transported parcels through the application of modern computer vision techniques can facilitate the timely and efficient detection of potential problems. This, in turn, can play a decisive role in ensuring the safety, security, and integrity of the cargo within the transportation network.

We first present a general and detailed literature review of computer vision applications in transportation logistics and warehousing. This review classifies the state-of-the-art and helps to identify open research questions. Afterwards, we focus on damage and tampering assessment for the use-case of last-mile delivery. This use-case imposes the necessity for highly flexible approaches with low hardware requirements, as the courier and end customer usually only have simple handheld devices such as smartphones at their disposal. Thus, in contrast to existing work that requires intricate multisensory setups and an a-priori known environment, all our approaches rely solely on a single RGB image as input.

The reliable detection and localization of parcels in images serves as a crucial foundation for downstream tasks dealing with damage and tampering assessment. Therefore, we first present a fully automated instance segmentation dataset generation pipeline to facilitate the training of targeted parcel detection modules. Due to the simplicity of the approach, datasets can be easily created and updated, and thus, robustness against potential appearance differences, e.g. across countries, can be achieved.

Subsequently, we present a novel pipeline for detecting tampering attempts, such as opening and resealing a parcel for unauthorized content removal. The newly developed pipeline first predicts the position of the eight parcel corner points and exploits this information to compute viewpoint-invariant frontal views for all visible side faces of the package at hand. By assuming that a reference texture is available, we are then able to reduce the problem of tampering detection to appearance change detection per parcel side surface. Appearance differences between two parcel side surface textures are recognized by visually aligning them through applying homogenization techniques and employing image similarity metrics in combination with thresholding.

Finally, we tackle the problem of damage detection and assessment for parcels, focusing on deformation damages only. We develop the novel targeted Artificial Neural Network (ANN) architecture CubeRefine R-CNN, which combines estimating a 3D bounding box with an iterative mesh refinement to adjust to deformations. Our approach simultaneously estimates the current, potentially deformed shape of a parcel and its original, pristine version. Thus, it enables a detailed damage assessment and quantification by directly comparing two 3D meshes. Moreover, we introduce Parcel3D, a novel synthetic dataset of damaged and intact parcels with full 2D and 3D annotations.

The performance of all approaches is evaluated quantitatively and qualitatively on real-world datasets. Furthermore, the datasets and source code have been published to ensure the reproducibility of our work.

Contents

Acronyms

AGV automated guided vehicle. 48, 55

AI Artificial Intelligence. 7

ANN Artificial Neural Network. vi, 7, 8, 12, 28, 52, 61, 80, 96, 97

AP Average Precision. 15, 16, 18, 19, 27–29, 31, 33, 37, 40, 72–74, 85–87, 106–108, 110, 111, 114

AR Augmented Reality. 43, 47, 48

AuC Area under the Curve. 15

CNN Convolutional Neural Network. iv, vi, 4, 8, 10, 11, 27, 30, 36, 37, 40, 43, 50, 68, 72, 73, 77, 79, 95, 96, 98, 103–112, 114

CW-SSIM Complex Wavelet Structural Similarity. 83

FP False Positive. 15

FPN Feature Pyramid Network. 29, 72, 74, 79, 85, 86, 93, 105, 106

GCN Graph Convolutional Neural Network. 12

GNN Graph Neural Network. 8, 11, 12, 33, 54

GSO Google Scanned Objects. 99–101

HCI Human-Computer-Interaction. 48

HOG Histogram of Oriented Gradients. 78, 83, 89

IoU Intersection over Union. 15, 16, 18, 30, 69

LLD Large Logo Dataset. 102

LPIPS Learned Perceptual Image Patch Similarity. 82, 89, 90, 94, 114

MAE Mean Absolute Error. 83, 89

MLP Multi-Layer Perceptron. 8–11

MS-SSIM Multiscale Structural Similarity. 82, 89

MSE Mean Squared Error. 13

NLP Natural Language Processing. 8

OCR Optical Character Recognition. 27, 28, 41

OKS Object Keypoint Similarity. 16

PMD Photonic Mixing Device. 52

RANSAC Random Sampling Consensus. 52

ReLU Rectified Linear Unit. 9

RFID Radio-Frequency Identification. 27, 32

ROC-AUC Area Under the Curve of the Receiver Operating Characteristic. 38

RPN Region Proposal Network. 104

SGD+M Stochastic Gradient Descent with Momentum. 72, 85, 105

SSIM Structural Similarity. 82, 89

SVM Support Vector Machine. 37, 52

ToF camera time-of-flight camera. 47, 49, 50, 53

TP True Positive. 15, 19

1 Introduction

Transportation logistics and warehousing are crucial parts of every supply chain and play an important role in the Industry 4.0 era [TV19]. However, companies working in the logistics sector are faced with a competitive environment and several challenges: customers demand faster, cheaper, more reliable and more precisely scheduled deliveries while shipping volumes can vary strongly. On top of that, traffic volumes in congested cities pose a problem and environmental concerns are becoming more relevant than ever. Due to the huge economic impact of transportation logistics and warehousing, it is crucial to cope with these challenges efficiently. Famously, FedEx founder Frederick W. Smith already knew the importance of data when saying "The information about the package is as important as the package itself" in the 1970s [Bal13]. While digitalization has been adopted in the logistics sector for many years now [Her+21], recent technological advancements, such as the broad availability of low-cost sensors, in combination with tremendous progress in the area of computer vision unveil an immense potential that might lift digitalization in the industry to a new level.

In the following, we motivate our problem setting in Section 1.1 and describe the objectives and limitations in Section 1.2. Subsequently, we summarize our contributions in Section 1.3 and present the outline of this thesis in Section 1.4.

1.1 Motivation

Process automation entails tremendous potential to tackle the challenges faced in the logistics sector [WRZ20]. While processes in transportation logistics and

warehousing have increasingly been digitalized over recent years, many labor-intensive, error-prone, and expensive tasks are still carried out manually or receive limited attention. One prime example of such processes regards safety and security considerations, which are of rising importance due to the increasing amount of valuable goods within supply chains [NZO18]. Especially damage and tampering assessment are necessary to guarantee process integrity and, thus, customer satisfaction. While it is not possible to reliably infer information on the state of the freight by mere inspection of its packaging, damage detection is vital to identify, analyze and prevent re-occurring patterns such as parcel deformations. Tampering detection, on the other hand, helps to guarantee the integrity of the freight by detecting potential manipulations (e.g. someone trying to hide that a parcel was unlawfully opened). Both of these problems are difficult to tackle since continuous visual monitoring along the supply chain is required. Within logistics facilities such as warehouses, it is possible to use intricate multisensory setups to address such issues. At the same time, however, companies deem the seamless integration into existing processes as vital to embrace novel technologies [NZO18]. To facilitate a broader industry adoption, it is thus, crucial to restrict additional, potentially obstructive hardware requirements by focusing on leveraging existing devices. In addition, in our considered scenario last-mile delivery, usually, only simple handheld devices are employed. Consequently, developing non-disruptive approaches for damage and tampering assessment in logistics and warehousing, which rely on simple sensor data such as a single RGB image only, is desirable.

1.2 Objectives and Limitations

This work strives to develop vision-based information retrieval approaches that help gather valuable information on the freight's identity, integrity, and shape for applications in transportation logistics and warehousing. Since in logistics, packaging goods is a standard procedure to handle, transport, and store goods safely and efficiently [Sag04], we focus on the arguably most common form of packaging: cuboid-shaped parcels. We seek to develop a robust approach

for parcel localization that does not rely on manually annotated training data. Using these localization capabilities, we aspire to design techniques for assessing damages and tampering of parcels. In this pursuit, we focus on the use-case of last-mile delivery, where only a simple handheld device is available. Thus, our approaches should prioritize flexibility and only require a single RGB image as input, which renders them applicable in various other scenarios as well.

The robustness of our parcel segmentation approach is considered w.r.t. the parcel's visual appearance, i.e. its texture and design. These can vary between countries and potentially evolve. Our approach strives to leverage image data from online search engines to be able to adapt to such changes without the necessity to manually label data. One limitation of relying on such data, however, is that copyright might apply, which can render it unusable for certain (commercial) applications.

Tampering assessment requires a two-step pipeline: we first need to reliably identify a parcel before searching for appearance changes on the packaging. We assume that parcel identification was performed in a previous step, e.g. by reading out the shipping label or by parcel texture-based matching [Cla+19]. Furthermore, we assume a database of the parcels' textures before tampering is available as a reference. Finally, we can only detect manipulation attempts that result in appearance changes in the parcel texture. If, for example, a parcel is opened carefully without damaging tape or packaging, our approaches will be unable to recognize this.

When treating damage assessment, we focus on shape deformation and do not consider other types of damages, e.g. caused by water. Also, only the packaging itself is analyzed, from which no reliable deduction on the state of the transported good is possible. Moreover, since we only rely on a single RGB input, we have no information on the (self-)occluded parts of the parcel and, consequently, a limited reconstruction accuracy. For the same reason, we can also not estimate scale reliably. Thus, without relying on a-priori known landmarks in the image, we cannot perform absolute volume estimations, which would be helpful for the downstream optimization of vehicle capacity usage.

1.3 Contributions

The main contributions of this thesis are:

- We review and categorize the literature regarding computer vision applications in transportation logistics and warehousing in depth.
- We present an approach for the automated generation of instance segmentation datasets, which leverages online image search engines. We investigate the importance of image selection and blending methods by carrying out a case study on parcel segmentation.
- Since no dataset of parcels with full 3D annotations is available, we introduce Parcel3D. It consists of more than 13 000 synthetic images of damaged and pristine parcels and provides 2D as well as 3D annotations.
- To treat tampering assessment, we propose a pipeline that combines parcel corner point detection with change detection approaches. Information on the parcel corners enables generating viewpoint-invariant parcel side surface representations, which facilitates detecting appearance changes.
- We develop an approach for single image 3D shape reconstruction of potentially damaged parcels, called CubeRefine R-CNN. Our approach simultaneously estimates an object's current, potentially deformed shape as well as its original pristine shape. Thus, we enable damage assessment by comparing 3D meshes, which allows detailed damage quantification.

The source code and datasets from our contributions are publicly available, and we provide an overview of all relevant resources in a separate Chapter, following the discussion in Chapter 7.

1.4 Outline

The remainder of this work is structured as summarized in Fig. 1.1 and outlined in the following. First, we briefly introduce the required fundamentals of computer vision and review the literature on computer vision applications in transportation logistics and warehousing in Chapters 2 and 3, respectively. Subsequently, we present an approach that leverages image data available online to increase the robustness of parcel recognition without the necessity of manually gathering and labeling training data in Chapter 4. We consider tampering assessment by combining parcel corner detection with change detection in Chapter 5. Finally, we treat damage assessment and quantification by lifting parcels from a single RGB image into 3D by estimating their full 3D shape in Chapter 6. Our work concludes with Chapter 7 by providing a discussion and outlining areas for future research.

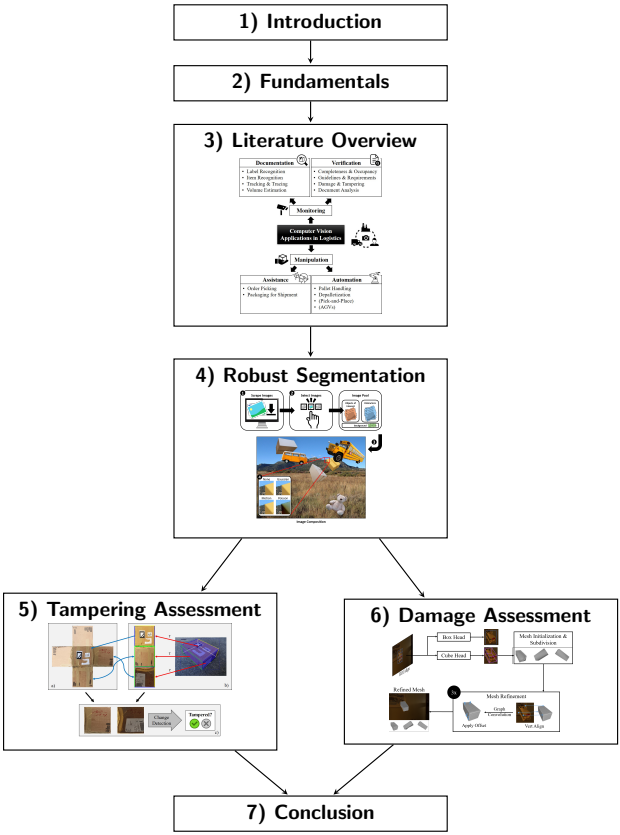


Figure 1.1: Visual overview of the structure of this thesis. [Sources of graphics: Chapter 3 from [Nau+23b], Chapter 4 from [Nau+22] ©2022 IEEE, Chapter 5 from [Nau+24] ©2024 IEEE, Chapter 6 from [Nau+23a] ©2023 IEEE]

2 Fundamentals

Nowadays, machine learning applications are ubiquitous and often utilize approaches that are categorized as deep learning. Machine learning intents to extract patterns from raw input data. Since most input modalities, such as images or text, are not trivial for machine learning algorithms to process, early works often relied on hand-engineered input features. These feature extraction approaches imposed a strong inductive bias towards what humans think could help the underlying task. As the field progressed and the deep learning era emerged, hand-engineered features were replaced by learned feature representations. These changes were enabled by the increase in data availability and computing resources [Sev+22], and advanced the state-of-the-art across many different tasks.

Since this work focuses on computer vision applications using deep learning in the context of logistics, we briefly introduce the fundamentals of deep learning and common computer vision tasks and metrics in Section 2.1 and Section 2.2, respectively. Schmidhuber [Sch15] presents a detailed historical overview of Artificial Neural Networks (ANNs) and we refer to Bishop [Bis06] for an in-depth introduction focusing on classical machine learning and to Goodfellow et al. [GBC16] for a detailed introduction into deep learning.

2.1 Introduction to Deep Learning

Artificial Intelligence (AI) has evolved from rule-based systems and classic machine learning to deep learning, which is now heavily used in research and industry. Apart from Reinforcement Learning, which is out of the scope of this work, the

literature broadly distinguishes between supervised and unsupervised learning. While the former assumes that the input data is coupled with associated target output values, the latter refers to approaches that do not rely on available target output information. Another distinction can be made w.r.t. the data modality that is used. While computer vision and Natural Language Processing (NLP) are arguably the most active areas of research, also tabular data, time series, and recommender systems are key applications of machine learning. It has also become common to mix different modalities to improve existing approaches or to enable tackling new types of problems [BAM19]. In the following, we focus on supervised learning and computer vision applications.

Multi-Layer Perceptrons (MLPs) are the basic building block of deep learning models and will be introduced in Section 2.1.1. Subsequently, we introduce Convolutional Neural Networks (CNNs) that can efficiently exploit the inherent structure of images in Section 2.1.2. Since this work aims to present novel models for 3D mesh reconstruction, Graph Neural Networks (GNNs), which operate over graphs, are introduced in Section 2.1.3. Finally, insights into the training procedure of ANNs are outlined in Section 2.1.4.

The introduction of MLPs, CNNs and the training procedure is based on Goodfellow et al. [GBC16]. Hamilton [Ham20] serves as the basis for the presented insights into GNNs.

2.1.1 Multi-Layer Perceptron

MLPs are fundamental and quintessential models in deep learning. They are frequently also called feedforward networks or feedforward neural networks and are crucial to machine learning practitioners. The terminology *neural* stems from the fact that they draw loose inspiration from neuroscience. An MLP tries to learn the best parameters $\boldsymbol{\theta}$ to approximate a function f^* through the mapping $y = f(\boldsymbol{x}; \boldsymbol{\theta})$ it defines. Here, $\boldsymbol{x}$ is a given input, and y is the corresponding desired output, which would correspond to a class label in the case of a classifier. MLPs are commonly composed of different functions, which can be connected

in a chain described by a directed acyclic graph. Exemplarily considering two functions $f^{(1)}$ and $f^{(2)}$, and chaining them leads to an MLP

$$f(\boldsymbol{x}) = f^{(2)}(f^{(1)}(\boldsymbol{x}))$$

with one hidden layer $f^{(1)}$ and one output layer $f^{(2)}$. If the two functions $f^{(i)},\ i \in \{1, 2\}$ were linear, this would limit the model's capacity to represent linear relationships only. To overcome this limitation, such linear models can be extended by applying the linear output function to a transformed input $\phi(\boldsymbol{x})$, with a nonlinear transformation. $\phi(\boldsymbol{x})$ provides a new set of features to describe the input $\boldsymbol{x}$ and is learned during training. A simple model is then described by

$$y = f(\boldsymbol{x}; \boldsymbol{\theta_1}, \boldsymbol{\theta_2}) = f^{(2)}(\phi(\boldsymbol{x}; \boldsymbol{\theta_1})) = \boldsymbol{\theta_2}^T \phi(\boldsymbol{x}; \boldsymbol{\theta_1}),$$

with parameters $\boldsymbol{\theta_1}$ and $\boldsymbol{\theta_2}$, which determine the novel representation of the input $\boldsymbol{x}$ and their linear mapping to the desired output, respectively. For the novel feature representation $\phi(\boldsymbol{x}; \boldsymbol{\theta_1})$, it is common to apply a nonlinear function to an affine transformation. The nonlinear function is called *activation function* and is typically applied element-wise. The Rectified Linear Unit (ReLU), which is defined as

$$g(z) = \max(0, z)$$

is arguably the most common choice for the activation function, although other nonlinear functions may be used. Finally, an MLP with one hidden layer, a ReLU activation function and the previously omitted additive bias terms can be defined as

$$f(\boldsymbol{x}; \boldsymbol{\theta_1}, \boldsymbol{b_1}, \boldsymbol{\theta_2}, b_2) = \boldsymbol{\theta_2}^T \max\left(0, \boldsymbol{\theta_1}^T \boldsymbol{x} + \boldsymbol{b_1}\right) + b_2$$

with parameters $\boldsymbol{\theta_1},\ \boldsymbol{\theta_2}$, and bias terms $\boldsymbol{b_1},\ b_2$ for the hidden and output layer, respectively.

2.1.2 Convolutional Neural Networks

CNNs are designed to operate on a known grid-like topology as for example encountered in image and time-series data. We follow the convention of referring to the operations utilized in CNNs by *convolution*, although it does not coincide precisely with the definition used in other fields such as pure mathematics or engineering. Commonly, the term *convolution* is used to refer to the related *cross-correlation* function, which is denoted by $*$ and defined as

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n),$$

where I is a two-dimensional image and K a two-dimensional kernel with index sets m and n. Such a discrete convolution can be represented by a matrix multiplication, which makes computation efficient.

This design of CNNs is influenced by neuroscientific principles and leverages (1) sparse intersections, (2) parameter sharing and (3) equivariant representations. (1) Sparsity is achieved by using a kernel that is smaller than the input image. This smaller kernel means that the output values of the convolution operation only depend on a subset of the input values, as opposed to MLPs, where each output is connected to all inputs. This, in turn, reduces the memory requirements and enhances its statistical efficiency by storing fewer parameters. For practical applications, well-performing kernel sizes are usually considerably smaller than the image dimensions, thus, rendering efficiency improvements quite significant. (2) In MLPs, each weight matrix element is used exactly once during the output calculation. In contrast to that, parameter sharing, as used in CNNs, enables the re-use of weights during the computation of the output of a layer by utilizing each kernel entry at every input position. This again reduces the memory requirements, however, the runtime of forward propagation remains unchanged. (3) Finally, the convolution operation is equivariant w.r.t. translation, which means that $f(g(x)) = g(f(x))$ holds for a convolution f and a translation g. Thus, if an object is moved in the input image, its feature representation in the output map of the convolution is moved by the same amount. This makes convolutions

very useful to detect common features, such as edges, which appear throughout the image. Note that convolutions are not equivariant w.r.t. other transformations such as scaling and rotation, and that they enable operating on inputs of variable size.

Apart from the convolutional operation, pooling layers are an integral component of CNNs. Pooling layers apply a so-called pooling function, which replaces an element with a 2D feature representation by summary statistics of nearby values. One common example is *max pooling* [ZC88], which returns the maximum value in a rectangular neighborhood.

For details on specific CNN architectures, we refer to the seminal works AlexNet [KSH12], ResNet [He+16], Mask R-CNN [He+17] and Vision Transformer [Dos+20] and to the review of Zhao et al. [Zha+19].

2.1.3 Graph Neural Networks

GNNs operate on graphs, which are a data structure that is encountered in numerous applications such as biology and social networks. Since 3D meshes are also naturally represented as graphs, GNNs are also important for 3D reconstruction, which is tackled within this work.

A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is composed of a set of nodes $\mathcal{V}$ and a set of edges $\mathcal{E}$, which describe the pairwise interactions between nodes. Each node $v \in \mathcal{V}$ and each edge $(u, v) \in \mathcal{E}$ can also have associated attribute or feature information. Tasks commonly tackled with a graph as the underlying data structure are node classification, relation prediction, community detection, graph classification and graph regression.

The key idea of GNNs is to exploit the graph structure and associated feature information in order to compute improved feature representations for nodes and edges. In contrast to images we discussed before, a graph has no predefined node ordering and the number of edges per node can vary. This means, that neither MLPs nor CNNs can operate on arbitrary graphs and thus, a novel type of deep

learning architecture is necessary. GNNs take a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with node features $\boldsymbol{X} \in \mathbb{R}^{d \times |\mathcal{V}|}$ as input and update the node embeddings $\boldsymbol{z}_u,\ \forall u \in \mathcal{V}$ by performing neural message passing, typically in an iterative fashion.

For a message passing step k, each node $u \in \mathcal{V}$ is considered and its hidden embedding $\boldsymbol{h}_u^{(k)}$ is computed by aggregating information from its neighborhood $\mathcal{N}(u)$:

$$\begin{aligned}\boldsymbol{h}_u^{(k+1)} &= \text{UPDATE}^{(k)}\left(\boldsymbol{h}_u^{(k)},\ \text{AGGREGATE}^{(k)}\left(\{\boldsymbol{h}_v^{(k)},\ \forall v \in \mathcal{N}(u)\}\right)\right) \\ &= \text{UPDATE}^{(k)}\left(\boldsymbol{h}_u^{(k)}, \boldsymbol{m}_{\mathcal{N}(u)}^{(k)}\right)\end{aligned}$$

with arbitrary differentiable functions UPDATE and AGGREGATE (e.g. ANNs) and the "message" $\boldsymbol{m}_{\mathcal{N}(u)}^{(k)}$. Since the AGGREGATE function operates on a set, it is permutation equivariant. Furthermore, through iterative application of the message passing, information from a larger neighborhood can be leveraged to enhance the node embeddings.

To use message passing in practice, it is of course necessary to define concrete instantiations for UPDATE and AGGREGATE. One very popular GNN architecture are Graph Convolutional Neural Networks (GCNs) [KW16], with the message passing function

$$\boldsymbol{h}_u^{(k)} = \sigma\left(\boldsymbol{\theta}^{(k)} \sum_{v \in \mathcal{N}(u) \cup \{u\}} \frac{\boldsymbol{h}_v}{\sqrt{|\mathcal{N}(u)||\mathcal{N}(v)|}}\right)$$

where σ is the activation function and $\boldsymbol{\theta}^{(k)}$ a trainable parameter matrix.

2.1.4 Training Procedure

Training ANNs relies on gradient-based optimization, where the goal is to find the best set of parameters $\boldsymbol{\theta}$ that result in a low cost function value $J(\boldsymbol{\theta})$. The cost function — in contrast to pure optimization — helps to indirectly optimize

the machine learning algorithm w.r.t. the performance measure of interest P. In practice, the cost function is usually defined as the average of some per-example loss L (e.g. the Mean Squared Error (MSE)) over the training set, i.e.

$$J(\boldsymbol{\theta}) = \mathbb{E}_{(\boldsymbol{x},y)\sim\hat{p}_{\text{data}}} L(f(\boldsymbol{x};\boldsymbol{\theta}), y)$$

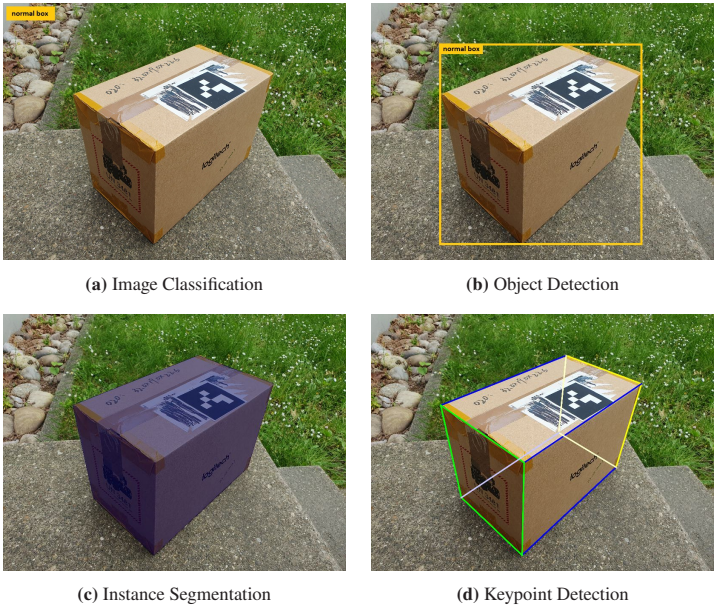
with input $\boldsymbol{x}$, ground truth output y, empirical data distribution $\hat{p}_{\text{data}}$ and predicted output $f(\boldsymbol{x};\boldsymbol{\theta})$. However, the final objective is to minimize the above cost function not for the empirical data distribution $\hat{p}_{\text{data}}$, but for the data generating distribution p_{data}. This corresponds to reducing the expected generalization error, also known as risk. Since the true data generating distribution $p_{\text{data}}(\boldsymbol{x}, y)$ is not known, we can only optimize for the empirical risk over m given training examples

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\hat{p}_{\text{data}}}[L(f(\boldsymbol{x};\boldsymbol{\theta}), y)] = \frac{1}{n}\sum_{i=1}^{m} L(f(\boldsymbol{x}^{(i)};\boldsymbol{\theta}), y^{(i)}).$$

This empirical risk minimization can be prone to overfitting if models have a capacity high enough to enable the memorization of the training set. Furthermore, in practical deep learning applications, the desired cost function can frequently not be optimized directly and a surrogate is used instead. One common example is the 0-1 classification loss, where the negative log-likelihood of the correct class is used as a surrogate. Two additional differences to classical optimization are that convergence criteria are used instead of halting at a local minimum and that the objective function is usually specified as the sum over per-example objective function values from the training set. The latter enables using only a subset of the available training examples, referred to as batch, for iterative gradient-based optimization.

2.2 Computer Vision Tasks and Metrics

Throughout this work, we will tackle several different tasks in computer vision. Thus, we introduce each such task in the following and explain the relevant



(a) Image Classification **(b)** Object Detection

(c) Instance Segmentation **(d)** Keypoint Detection

Figure 2.1: Overview of common 2D computer vision tasks.

evaluation metrics inspired by Szeliski [Sze22]. First, we focus on computer vision tasks in 2D in Section 2.2.1, and subsequently, we introduce the relevant tasks and metrics for 3D image understanding in Section 2.2.2.

2.2.1 2D Image Understanding

Images reduce our 3D world to a 2D representation. They are able to efficiently capture information about our environments, which would be difficult to describe by using other modalities, such as text. Analyzing images has a myriad of applications, and can be categorized into different tasks, which we summarize in Fig. 2.1 and introduce briefly below. For a more detailed introduction, we refer to Szeliski [Sze22].

Image Classification

Summarizing a whole image by assigning it a single class label is called image classification. While images oftentimes do not only represent one single object, the task is to identify the predominant class. We visualize an example in Fig. 2.1a.

Object Detection

Images can contain multiple objects and the object positions may vary strongly. Object detection refers to the task of localizing an object by providing a 2D bounding box and classifying it. An example output is visualized in Fig. 2.1b.

The Jaccard Index [Jac01] or pixel-wise Intersection over Union (IoU) is commonly used to assert the quality of a detected bounding box. It is defined as the quotient between the intersection and the union of the predicted and the ground truth bounding box. To compute a single metric over a set of images, and thus, to assess the quality of an object detector, commonly Box AP is used. For its calculation, an IoU threshold τ is chosen, and the Area under the Curve (AuC) of the Precision-Recall curve is computed by iterating over all detections sorted by detection confidence and classifying them as True Positive (TP) or False Positive (FP). Box AP_{50} refers to the Average Precision (AP) at an IoU of $\tau = 50\%$ and Box AP refers to the mean Box AP which is computed as the average over Box AP_τ $\forall \tau \in [0.50, 0.55, 0.60, \cdots, 0.95]$.

Instance Segmentation

Instance segmentation goes beyond bounding boxes and provides pixel-wise classification information on an instance basis. The result is a segmentation mask for each object instance in the image, as visualized in Fig. 4.3c.

The primary metric to evaluate instance segmentation algorithms is Mask AP. Its calculation is equivalent to Box AP, where the IoU is calculated by comparing the predicted and the ground truth segmentation mask instead of bounding boxes.

Keypoint Detection

Many object categories can also be described by a set of important keypoints. A common example are human faces, where the position of the eyes, nose, mouth, etc. are relevant landmarks. In this work, we use the vertices of a 3D bounding box as keypoints to describe parcels, as visualized in Fig. 2.1d.

A keypoint k_i is defined as $[x_i, y_i, v_i]$, where x_i and y_i are the positions in the image and

$$v_i = \begin{cases} 0 & \text{if keypoint is not labeled,} \\ 1 & \text{if keypoint is labeled and not visible,} \\ 2 & \text{if keypoint is labeled and visible} \end{cases}$$

indicates the visibility. To evaluate the keypoint detectors Object Keypoint Similarity (OKS) instead of IoU is used.[1] OKS is defined as

$$\text{OKS} = \sum_i \frac{\exp\left\{-\frac{d_i^2}{2s^2\kappa_i^2}\right\} \cdot \delta\left(v_i > 0\right)}{\sum_i \delta\left(v_i > 0\right)}$$

for all keypoints k_i and the Dirac delta function δ. d_i is the Euclidean distance between the predicted and ground truth keypoint position, s the square root of the object's area and $\kappa_i = 2\sigma_i$ with per keypoint variance $\sigma_i^2 = \mathbb{E}(d_i^2/s_i^2)$. Note that the per keypoint variance is computed from redundant human annotations. Finally, Keypoint AP is computed equivalently to Box AP and Mask AP, however, by setting an OKS instead of an IoU threshold.

[1] We follow the definition from https://cocodataset.org/#keypoints-eval. [Last accessed on Sept. 20, 2024]

(a) Input Image

(b) 3D Bounding Box Detection

(c) 3D Reconstruction

Figure 2.2: Overview of common 3D computer vision tasks.

2.2.2 3D Image Understanding

Through the reduction of our 3D world to a 2D image, we loose valuable information about object positions, sizes and shapes. In the following, we present tasks that try to lift the available 2D image information back to three dimensions. More precisely, we will introduce 3D bounding box detection and 3D reconstruction from a single RGB image as input. An overview is presented in Fig. 2.2.

3D Bounding Box Detection

Equivalently to bounding box detection in 2D, finding the 3D bounding box is an important task (cf. Fig. 2.2b). It is frequently tackled in the context of autonomous driving [Ma+23], but also has applications in other domains [Nau+23a]. When

considering the 2D projection of a parcel, the task closely relates to keypoint detection.

To evaluate 3D bounding box detectors, we follow Brazil et al. [Bra+23] and use the AP of the 3D IoU, called AP3D. This again corresponds to computing Box AP, however, using the 3D bounding box IoU as metric. Since the IoU in 3D drops significantly faster than in 2D, AP3D is computed as the average over AP3D_τ $\forall \tau \in [0.05, 0.10, 0.15, \cdots, 0.50]$.

3D Reconstruction

Inferring the full 3D shape of an object (e.g. in the form of a mesh) is a common task in computer vision and visualized in Fig. 2.2c. While we focus on approaches using only a single RGB image, multiple RGB images or point clouds are also common input modalities. Comparing shapes in 3D is difficult since 3D bounding box or voxel IoU are often poor shape similarity indicators [Tat+19]. Instead, independent of the shape representation, frequently point cloud-based metrics are used. Chamfer distance is a common metric, which describes the general alignment of two point clouds X, Y by comparing respective nearest points and is defined as

$$d(X,Y) = \frac{1}{|X|} \sum_{\boldsymbol{x} \in X} \min_{\boldsymbol{y} \in Y} \|\boldsymbol{x} - \boldsymbol{y}\|^2 + \frac{1}{|Y|} \sum_{\boldsymbol{y} \in Y} \min_{\boldsymbol{x} \in X} \|\boldsymbol{x} - \boldsymbol{y}\|^2.$$

Furthermore, the normal consistency NC is a commonly employed metric and is defined as

$$\begin{aligned} \text{NC}(X,Y) = 1 - \frac{1}{|X|} \sum_{\boldsymbol{x} \in X} & \left| \boldsymbol{n}_x \cdot \boldsymbol{n}_{\arg\min_{\boldsymbol{y} \in Y} \|\boldsymbol{x} - \boldsymbol{y}\|} \right| \\ & - \frac{1}{|Y|} \sum_{\boldsymbol{y} \in Y} \left| \boldsymbol{n}_{\arg\min_{\boldsymbol{x} \in X} \|\boldsymbol{x} - \boldsymbol{y}\|} \cdot \boldsymbol{n}_y \right| \end{aligned}$$

where $\boldsymbol{n}_i$ is the normal vector corresponding to point $\boldsymbol{i}$.

Finally, Mesh AP [GMJ19] provides a metric summarizing the overall reconstruction quality, similar to Box AP and Mask AP in 2D. It is based on the F1-score, i.e. the harmonic mean of precision and recall

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

When comparing a ground-truth pointcloud X with a predicted pointcloud Y using the F1-score, we need to define a distance threshold ψ, and F1_ψ is computed using

$$\text{Precision}_\psi = \frac{1}{|Y|} \sum_{\boldsymbol{y} \in Y} \delta \left(\min_{\boldsymbol{x} \in X} \|\boldsymbol{x} - \boldsymbol{y}\|^2 \leq \psi \right)$$

and

$$\text{Recall}_\psi = \frac{1}{|X|} \sum_{\boldsymbol{x} \in X} \delta \left(\min_{\boldsymbol{y} \in Y} \|\boldsymbol{x} - \boldsymbol{y}\|^2 \leq \psi \right)$$

with the Dirac delta function δ. Mesh AP_τ is then defined as the AP where F1_ψ is used as metric and τ as threshold. More precisely, a detection is TP when $\text{F1}_\psi > \tau$, it is not a duplicate and the predicted label is correct [GMJ19].

3 Literature Overview

The intersection of computer vision and logistics is and has been an active area of research. We present a detailed overview of existing literature categorized from an application-oriented view as summarized in Fig. 3.1 and available online at https://a-nau.github.io/cv-in-logistics. We start in Section 3.1 by reviewing the literature on monitoring, where applications considering image-based information retrieval from the environment are presented. Subsequently, in Section 3.2, manipulation applications are introduced, where in addition to information retrieval, the interaction with the environment plays a key role. Next, in Section 3.3 the literature is categorized from a computer vision point of view, and overviews of datasets and industrial solutions are presented. Finally, we discuss the results of our literature review w.r.t. the objectives of this thesis in Section 3.4.

Sections 3.1 to 3.3 have been previously published as a preprint and are direct quotes from Naumann et al. [Nau+23b], including tables and figures. These sections are marked with ❞ [Nau+23b] in the respective headline.

3.1 Monitoring ❞ [Nau+23b]

Monitoring refers to observing processes in order to retrieve valuable information. The retrieval of information is based on visual perception. We distinguish two different cases: (1) Documentation, where we only want to retrieve information to document processes. Details are presented in Section 3.1.1 and summarized in Table 3.1. In this context, data is collected and stored, but not used as the basis for subsequent process automation. (2) Verification, where the retrieved

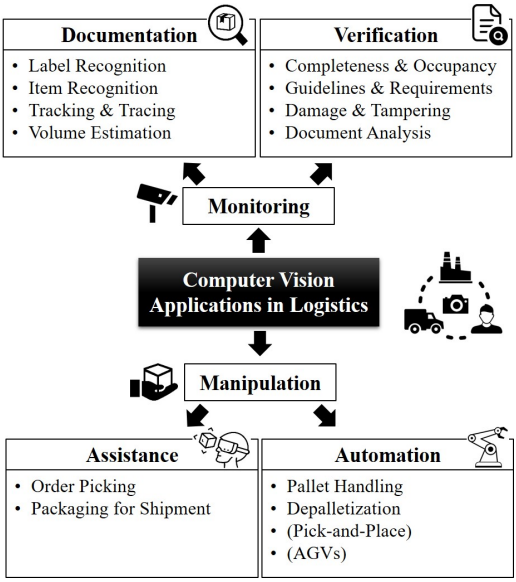


Figure 3.1: Overview of literature categorization. [Graphic from [Nau+23b]]

information is used to verify certain assumptions (e.g. by comparing with a database of shipments). The relevant literature is discussed in Section 3.1.2 and summarized in Table 3.2. Note, that this application-oriented categorization causes large overlaps w.r.t. the computer vision approaches that are used and a separate categorization w.r.t. computer vision approaches will be presented in Section 3.3.1.

Table 3.1: Overview of the literature on Documentation (cf. Section 3.1.1).

Paper	Summary	Application	Data Type	Public Dataset	Approach Type	Objects
[BBS21]	Detection of dangerous goods labels and volume estimation by pointcloud segmentation	Depalletization, Label Recognition, Volume Estimation	RGB, RGB-D, Real, Synthetic		Deep Learning	Label, Pallet, Parcel
[Mis+19]	Low cost embedded vision system for the detection of 1D barcodes	Label Recognition	RGB		Classical Approach	Label
[Dör+19]	Training data generation for shipping label recognition	Label Recognition	RGB, Synthetic		Deep Learning	Label
[Suh+19]	Label recognition by first detecting barcodes and then using this information for angle calibration	Label Recognition	RGB		Classical Approach, Deep Learning	Label
[BSB20]	Detecting barcodes in images	Label Recognition	RGB, Synthetic		Deep Learning	Label
[WCM22]	Review on barcode detection	Label Recognition				Label
[Kam+22]	Comparison of CNNs for barcode detection	Label Recognition	RGB, Real		Deep Learning	Label, Parcel
[MGF20]	Synthetic dataset generation for applications in logistics	Item Recognition	RGB, Synthetic		Deep Learning	Small Load Carrier
[May+20]	Dataset for item recognition in logistics contexts	Item Recognition	RGB, Real	✓	Deep Learning	Multiple
[Nau+20]	Parcel side surface segmentation by exploiting plane detection	Item Recognition	RGB, Real	✓	Deep Learning	Parcel
[Nau+22]	Automated instance segmentation dataset generation applied to parcel logistics	Item Recognition	RGB, Synthetic		Deep Learning	Arbitrary, Parcel
[She+12]	Framework for OCR on containers	Item Recognition	RGB			Container/ Trailer

Table 3.1 (continuation): Overview of the literature in Documentation (cf. Section 3.1.1).

Paper	Summary	Application	Data Type	Public Dataset	Approach Type	Objects
[Dör+20b]	Localization of pallets and the analysis of their composition	Item Recognition, Verify Completeness	RGB, Real		Deep Learning	Pallet, Small Load Carrier
[Dör+21]	Localization of pallets and the analysis of their composition	Item Recognition, Verify Completeness	RGB, Real		Deep Learning	Pallet, Small Load Carrier
[Wei+10]	Continuous detection, localization, and identification of parcels and bins in logistics processes	Tracking and Tracing	RGB		Classical Approach, Fiducial Markers	Container/ Trailer, Parcel
[Bor+14]	System for pallet monitoring	Tracking and Tracing	RGB, RGB-D, Real		Classical Approach, Fiducial Markers	Pallet
[Cla+19]	Industry scale approach for tracking parcels in a logistics facility.	Tracking and Tracing	RGB, Real		Deep Learning, Fiducial Markers	Parcel
[Hu+21a]	Tracking parcels inside a moving truck	Tracking and Tracing	RGB-D, Real		Deep Learning	Parcel
[Rut+21]	Re-identification for chipwood pallet blocks of Euro pallets	Tracking and Tracing	RGB, Real	✓	Deep Learning	Pallet
[Klü+22]	Re-identification for chipwood pallet blocks of Euro pallets	Tracking and Tracing	RGB, Real		Deep Learning	Pallet
[Rut+22a]	Re-identification for chipwood pallet blocks of Euro pallets	Tracking and Tracing	RGB, Real	✓	Deep Learning	Pallet
[Bor+13]	Prototyped solutions for volume scanning using two MS Kinect cameras	Volume Estimation	RGB-D, Real		Classical Approach	Pallet, Parcel
[LWP13]	Estimating the volume of a single parcel using a mobile device	Volume Estimation	RGB, Real		Classical Approach, Pattern Matching	Parcel, Parcel
[Kuc+19]	Dimension estimation of static objects for logistic applications	Volume Estimation	RGB-D, Real		Classical Approach	Multiple, Parcel

Table 3.2: Overview of the literature on Verification (cf. Section 3.1.2).

Paper	Summary	Application	Data Type	Public Dataset	Approach Type	Objects
[NZO18]	Damage and tampering detection in a postal security framework from multiple cameras	Damage and Tampering Detection	RGB, Real		Classical Approach	Parcel
[Mal+21]	Concept for damaged parcel detection with CNNs	Damage and Tampering Detection				Parcel
[Nau+23a]	Damage assessment using single image 3D reconstruction	Damage and Tampering Detection	RGB, Real, Synthetic	✓	Deep Learning	Parcel
[Dör+20b]	Localization of pallets and the analysis of their composition	Object Detection, Verify Completeness	RGB, Real		Deep Learning	Pallet, Small Load Carrier
[Dör+21]	Localization of pallets and the analysis of their composition	Object Detection, Verify Completeness	RGB, Real		Deep Learning	Pallet, Small Load Carrier
[ÖAN16]	Recognize the occupancy status of the load handling device of forklift trucks	Verify Occupancy	RGB, Real		Classical Approach, Fiducial Markers	Fork Lift
[Li+21]	Recognize congestions on conveyor belts	Verify Occupancy	RGB, Real		Classical Approach	Conveyor Belt, Parcel

3.1.1 Documentation

The documentation of business processes is very important, especially in complex transportation networks, as frequently encountered in supply chains. To simplify the documentation of logistics processes, such as verifying incoming goods, several directions have been suggested in research. The emphasis lies on automated vision-based information retrieval, and we present works on four different documentation tasks: label recognition (Section 3.1.1.1), item recognition (Section 3.1.1.2), tracking and tracing (Section 3.1.1.3), and volume estimation (Section 3.1.1.4). For each section, we first describe the task and present an overview of the existing literature. Subsequently, we summarize the findings, and briefly suggest further research directions. Additionally, an overview of the literature reviewed in this section is presented in Table 3.1.

3.1.1.1 Label Recognition

Transport labels uniquely identify shipments and thus, are fundamental for the organized management of goods. Especially at every intersection point of the supply chain where goods are transferred, label detection is important to verify the completeness of the shipment. Apart from transport labels, also other types of labels can contain relevant information that should be retrieved. One such example are dangerous goods labels, which are crucial for the safety of operators and cargo.

Mishra et al. [Mis+19] present a low-cost embedded vision system for the detection of 1D barcodes. They use traditional image processing to detect and rotate the barcode. They do not present results on the accuracy of the detections, however, focus on the execution time.

Dörr et al. [Dör+19] present different approaches to generate a targeted dataset for logistics transport label detection. They take images of load carriers in realistic environments, where the load carriers have a colorful and easy-to-segment sheet of paper attached, where they would usually have a transport label. This enables

them to easily paste real transport labels onto these colorful dummy labels. They investigate the trade-off between realism and randomness and find that accurate object detection models can also be trained on synthetic data only. Contrary to human intuition, realism is not always advantageous, but using randomized backgrounds can yield good results. The authors report Box APs between 0.65 and 0.92 for different scenarios.

Suh et al. [Suh+19] develop a label recognition pipeline that first detects barcodes on the shipping label, and then uses this information for angle calibration. The angle calibration horizontally aligns the barcodes, and thus also the whole visible label. Afterwards, they go beyond label detection and use a second Convolutional Neural Network (CNN) to detect the bounding box around the address. Finally, Optical Character Recognition (OCR) is employed to extract the address as text. The authors evaluate the barcode and address recognition accuracy and reach $94.7\,\%$ and $93.62\,\%$ respectively, while allowing random label rotation of up to $20°$.

Focusing on maritime logistics, Shetty et al. [She+12] tackle Container OCR. They focus on port logistics and present a framework composed of a container detection module, a decision engine, and a central risk management system. The container detection module comprises an OCR module for container code retrieval which can be fused with Radio-Frequency Identification (RFID) information of cranes and other equipment to increase robustness. The paper suggests a framework and does not present empirical results or concrete implementations for the modules.

Brylka et al. [BSB20] revise the problem of detecting barcodes in images. In contrast to prior approaches, they tackle multiple real-world issues at the same time, such as poor illumination, noise, and motion blur. Their approach consists of four consecutive steps: (1) localization, (2) segmentation and contour estimation, (3) orientation estimation and contour refinement, and (4) decoding with optional deblurring. Modern CNNs are used, which is the reason for generating synthetic training datasets for each subtask. They present two datasets for barcode localization, each comprising 25,000 images. For segmentation, they generate 60,000 images, and finally, for deblurring a dataset of 300,000 images. The evaluation

is performed on a dataset of 400 real images, where each image contains several of 30 different barcodes. The dataset is manually annotated and named AWB dataset. They report a recall of 0.446 and a precision of 1.0 for the full approach.

Brylka et al. [BBS21] also tackle the problem of label detection, however, since their focus is on dangerous good, we review their approach in Section 3.1.2.2 on verifying guidelines and requirements.

Kamnardsiri et al. [Kam+22] perform a case study by analyzing five different Artificial Neural Network (ANN) architectures for 1D barcode detection. They present two new datasets: InventBar and ParcelBar with 527 and 844 images, respectively. Additionally, the evaluation considers several existing datasets. Results show that YOLOv5 [Joc+22] performs best with an AP of 91.3 while YOLOx [Ge+21] is the fastest model considering the average runtime for all experiments.

Summary Label recognition mostly focuses on barcodes [Mis+19; Suh+19; BSB20; Kam+22], where also complex environments have been investigated. For a more in-depth literature review of barcode detection, we refer to the survey of Wudhikarn et al. [WCM22]. In addition, the detection of dangerous goods labels [BBS21], transport label detection [Dör+19] and container OCR [She+12] have been tackled.

Outlook Label detection requires object detection or semantic segmentation algorithms. Both fields are very active areas of research [Zou+23; Min+22]. Advances in these areas can be leveraged to improve accuracy, train with less data and increase robustness in difficult scenarios. Especially barcode detection has been studied thoroughly and numerous datasets are publicly available. While Kamnardsiri et al. [Kam+22] performed an analysis for a selection of algorithms, it would be interesting to analyze more diverse scenarios similar to Brylka et al. [BSB20]. Other fields, apart from barcode detection, lack the availability of diverse datasets and the effective use of synthetic data can be investigated.

3.1.1.2 Item Recognition

We use item recognition in this context to refer to localizing and classifying relevant logistics objects or items in an image. Computer vision taxonomy distinguishes object detection (cf. Zhao et al. [Zha+19] and Zou et al. [Zou+23]) and semantic segmentation [Min+22], however, in our application-oriented context no such distinction is made. Recognizing items is very helpful for documentation, e.g. to identify or count incoming or outgoing goods.

Mayershofer et al. [May+20] present the LOCO dataset which consists of 39,101 images in logistics environments, of which 5,593 images are annotated with bounding boxes. Annotations are performed manually for five logistics-specific object categories: small load carrier, pallet, stillage, forklift and pallet truck. The annotations are very unbalanced, since there are roughly 120,000 annotations for the class pallet, however less than 25,000 annotations for the second most frequent class small load carrier. More specifically, the super-category load carrier, which includes pallet, small load carrier and stillage has 43 times more annotations than the super-category transportation vehicles, which includes pallet truck and forklift. They report a Box AP_{50} of 20.2 using a ResNet-50-FPN [He+16; Lin+17] when fine-tuning on LOCO. Extensions to the dataset are planned, which include annotating more objects, incorporating 3D data and also providing segmentation masks.

Another logistics-specific dataset was presented by Mayershofer et al. [MGF20]. The dataset contains synthetic training data and an industrial evaluation dataset that comprises 1460 manually annotated images and focuses on five different types of small load carriers, as they are standardized by the VDA[1]. The synthetic training data is created by choosing a random image as floor and dropping distractor objects, as well as the objects of interest onto this floor in a physics-based simulation. Blender[2] is used, and since small load carriers are often stacked also the relation

[1] Verband der Automobilindustrie e.V., see https://www.vda.de/en [Last accessed on Sept. 20, 2024].

[2] See https://www.blender.org [Last accessed on Sept. 20, 2024].

between them is modeled. The synthetic dataset enables application for the real-world use-case of small load carrier detection, however, the detection quality is lower due to the domain gap. The authors train Yolov3 [RF18] report a recall of 0.42 and a precision of 0.4 at an Intersection over Union (IoU) of 50%.

Naumann et al. [Nau+20] tackle the problem of parcel segmentation and focus on the segmentation of all its side surfaces. Additionally having plane-level segmentation information facilitates the comparison of parcel photos that were taken from different angles (e.g. for tampering detection) since it allows normalizing each side surface view by applying a projective transformation. In contrast to the other approaches, no task-specific dataset is needed. Their method combines modern approaches for plane segmentation [Liu+19] that were trained on indoor room data with approaches for contour detection [Can86; SRS20]. In this way, the approach is conditioned to focus on the geometry information and is less influenced by appearance changes, e.g. through different parcel colors. The authors report an average IoU over all segmentation masks of 85.2.

While the previous works only used RGB images, Fontana et al. [FZL21] present an approach for parcel detection based on RGB-D data. They compare an approach that combines a Mask R-CNN [He+17] with post-processing to a clustering approach on the depth data. The clustering assumes prior knowledge of the box sizes, which limits its generalizability. Both approaches show similar performance (errors of around $5\,\mathrm{mm}$ and $1°$), while the learned method is slightly better.

Naumann et al. [Nau+22] present work on automated instance segmentation dataset generation. They present a case study for parcels and automatically scrape relevant image data from the internet. In the first step, only images with a homogenous background are kept and a class agnostics background removal technique is applied. Afterwards, three different image selection processes are analyzed that are based on mask convexity, CNN inference, and manual selection. Finally, the dataset is created by randomly pasting objects of interest together with distractor objects onto random backgrounds, similar to Dwibedi et al. [DMH17]. Results show that for the case study, manual selection of relevant parcel instances

is not superior to simple pre-processing based on mask convexity (Mask AP 81.5 vs 86.2).

Summary Item recognition has been investigated for parcels [Nau+20; FZL21; Nau+22], small load carriers [MGF20; Dör+21] and general logistics objects such as pallets and forklifts [May+20]. Note, that other approaches also rely on object detection as part of their pipeline, however, this section focused on research where object detection is the main point of interest.

Outlook Similar to Section 3.1.1.1 the computer vision tasks object detection and instance segmentation are relevant. Both fields are very active areas of research [Zou+23; Min+22], and advances can enable improvements in accuracy, training with less data and increasing robustness in difficult scenarios. In general, sufficient and high-quality data frequently is a limiting factor, which leaves an opportunity for contributions. To improve the transfer from synthetic training data to real-world applications, generative approaches such as Generative Adversarial Networks [Goo+14] and diffusion models [Rom+22] can be used.

3.1.1.3 Tracking and Tracing

Logistics objects move along supply chains that usually comprise several parts. Thus, in order to analyze the trajectory of specific items, it is necessary to be able to track them across these stages. Tracking can refer to internal tracking (e.g. within a facility) or global tracking (e.g. across companies and facilities). Especially for dangerous goods or in case an item gets lost, the information on the prior trajectory of an item is crucial to identify its whereabouts and thus, to guarantee operator safety and customer satisfaction.

Weichert et al. [Wei+10] consider the continuous detection, localization, and identification of parcels and bins in logistics processes. They consider roller and conveyor belt systems as typical representatives for automated transportation systems and suggest moving away from sensors such as light barriers and barcode

readers and substituting them with low-cost cameras and RFID systems. The combination of low-cost cameras and RFID systems allows one to identify an item either by detecting the marker with image processing or by reading out the data stored on the RFID tag. The authors present case studies on the influence of the camera position, camera type, marker type and object distance on the marker detection accuracy. They find a close camera position and high image resolution beneficial, while there is no clear winner for the marker type.

Borstell et al. [Bor+14] present a system for pallet monitoring. They use a heterogeneous sensor setup and a system architecture with subsystems for pallet identification, pallet dimensioning, vehicle positioning and load change detection.

Clausen et al. [Cla+19] present an industry-scale approach for tracking parcels on conveyor belts in a logistics facility. They use a Siamese network [Bro+93] for parcel re-identification, to which they add a fully connected network with only one layer consisting of 1024 neurons. A manually labeled dataset of 3,306 images from 37 different cameras, which contain a total of 14,248 parcels was created. In addition to that, they present a calibration approach. Instead of manually calibrating the multi-camera framework, a single drive-by of a calibration parcel is enough for each conveyor belt. They present extensive evaluations, also comparing to classical tracking approaches and show the superiority of their approach. Currently, around 81% of parcels are tracked correctly, while they name human interaction as a main cause of failure.

Noceti et al. [NZO18] also tackle the problem of tracking and tracing, however, since their focus is on damage and tampering detection, we review their approach in Section 3.1.2.3.

Hu et al. [Hu+21a] address the problem of tracking parcels inside a moving truck. A multi-RGB-D camera setup is used to overcome the limited field of view and occlusions of a single camera setup. They present a new calibration procedure using a sphere. After calibration, their approach first creates a unified scene by merging the available RGB-D information. On the resulting pointcloud, an approach for segmenting rectangular planes is applied. These planes are used to find and evaluate box candidates. To track a parcel, the position of its centroid,

its size, and its rotation relative to the reference coordinate system is used. They collect their own dataset with cuboids in different arrangements and report an average detection rate of 0.976 while enabling real-time usage.

Rutinowski et al. [Rut+21] tackle the problem of re-identification for chipwood pallet blocks of Euro pallets. For this purpose, they create a dataset consisting of 502 different pallet blocks and a total of 5,020 images. The authors compare different architectures for the task and find a competitive algorithm for person re-identification [Sun+18] as the most suitable. They report an AP of $98\,\%$ and a matching accuracy of $97\,\%$.

Klüttermann et al. [Klü+22] use the dataset from [Rut+21] and present first results using anomaly-based re-identification of pallet blocks. The authors identify anomalies by computing descriptive statistics of 16×16 image patches. Detected anomalies are combined into a graph, thus, reducing the size of pallet block representation. These graphs are processed by a Siamese Graph Neural Network (GNN) to map them into an embedding space. Finally, in order to retrieve the matching pallet block for a given input, its nearest neighbor in the embedding space is used. The accuracy of the approach is currently not competitive and is reported with $27\,\%$.

Rutinowski et al. [Rut+22a] also tackle the problem of re-identification for Euro pallets. They present a new dataset consisting of 32,965 pallet blocks. Of each pallet block four images are taken, resulting in a total of 131,860 images. The dataset was generated automatically by monitoring conveyor belts in the warehouses of two German companies. Similar to [Rut+21], they apply a Part-based Convolutional Baseline (PCB) network [Sun+18] and report an accuracy of $98\,\%$.

Summary Tracking and tracing have been investigated for parcels [Wei+10; NZO18; Cla+19; Hu+21a], and pallets [Bor+14; Rut+21; Rut+22a; Klü+22]. Approaches are mostly vision-based, partially rely on RGB-D imagery [Hu+21a] and leverage literature from person re-identification [Rut+22a]. Moreover, whole

systems for automating such processes [Bor+14; Rut+22b] and the monitoring of logistics road traffic [BTL19] have been investigated.

Outlook For the re-identification of parcels the approach of Rui et al. [Rui+20] seems promising, as it is tailored towards cuboid-shaped objects. Moreover, advances in feature matching [Sar+20] could be leveraged for re-identification. For a review on re-identification, we refer to Khan et al. [KU19] and Ye et al. [Ye+22].

3.1.1.4 Volume Estimation

Volume estimation refers to computing the volume of a single item or a set of items. It is especially relevant to extract this information for optimizing downstream tasks, such as container loading.

DHL developed two prototyped solutions for volume scanning using two MS Kinect cameras in 2013 [Küc13; Bor+13]. The first solution is a gate approach while the second mounts cameras directly on a forklift truck mast. The volume is estimated by confining the overall pointcloud to a dedicated area and summing over a discretized height map. Their system only needs $250\,\mathrm{ms}$ for the capturing process to minimize the idle time during measurement. Furthermore, they calibrate their system to achieve more accurate results. They discretize the floor plane into tiles and estimate the height for each of them. The final volume can then easily be computed by summing up over all tiles. They analyze and compare different configurations that vary w.r.t. the relevant area and the camera setup. The accuracy of their extent estimation ranges between 10 and $13\,\mathrm{mm}$ for the considered scenarios.

Laotrakunchai et al. [LWP13] present an approach for estimating the volume of a single parcel using a mobile device. Their approach utilizes two different data modalities. They use the cell phone acceleration sensor to measure the parcel extents by dragging the cell phone along its dimensions. This information is complemented with two images (start and end of dragging) to enable measuring

parcel extends from a distance. For each image, the object region is selected manually and SURF [Bay+08] keypoints and descriptors are used for feature matching and subsequent disparity computation. The final result is retrieved by applying a Gaussian weighted interpolation scheme. Four different datasets are collected and the performance is analyzed for different object distances. The average percentage error at distances of $1\,\mathrm{m}$, $1.5\,\mathrm{m}$, $2\,\mathrm{m}$, $2.5\,\mathrm{m}$ and $3\,\mathrm{m}$ is $10.2\,\%$, while no clear trend on the dependence of the distance is present.

In addition to the detection of dangerous labels as will be presented in Section 3.1.2.2, Brylka et al. [BBS21] also treat the problem of volume estimation. They generate a dataset by using a setup with multiple depth cameras and fiducial markers. This way, they automatically annotate 150 pallets with parcels that are always brown boxes and have 10 different dimensions. They train BoNet [Yan+19] and evaluate it on a separate validation dataset. No details on the validation dataset or quantitative results are given. The presented qualitative results look promising.

Kucuk et al. [Kuc+19] develop a system for dimension estimation of static objects for logistic applications. An RGB-D camera is employed to capture a pointcloud of the object. Without going into details, the authors name spatial and temporal filters as post-processing of the pointcloud and do not report on the method used for finding the minimum bounding box for the object. They perform evaluations on a range of objects, such as cylinders, tubes and cubes and report an error of less than $0.5\,\mathrm{cm}$ in each dimension under good lightning conditions and a suitable exposure time.

Further specialized approaches include [ST14], which heavily relies on manual input and [Sun+20], which only evaluates on four different parcels.

Summary Volume estimation is mostly tackled using one [Kuc+19] or multiple [Küc13; Bor+13; BBS21] RGB-D cameras. In addition to that, Laotrakunchai et al. [LWP13] investigated the usage of cell phones leveraging their acceleration sensor.

Outlook Since RGB-D data is used frequently, studying the capabilities of pointcloud classification and pointcloud segmentation algorithms, as reviewed by Grilli et al. [GMR17] and Bello et al. [Bel+20], seems promising. New applications that, to the best of the authors' knowledge, have only been investigated commercially, can be considered. Examples include measuring the load volume on a driving forklift. For applications in scenarios with limited sensor availability, e.g. during last-mile delivery, also approaches for single RGB shape reconstruction and volume estimation are interesting. Related methodological literature is reviewed in [Kha+22]. One very promising approach which has been used for shape reconstruction [Nau+23a] is Cube R-CNN [Bra+23]. By providing suitable data during training, it can also estimate scale and thus, volumes independent of otherwise necessary scale landmarks.

3.1.2 Verification

Verification, in contrast to documentation, does not only encompass the mere retrieval of information but at the same time compares it to existing data. In the following, we analyze approaches for checking completeness and occupancy (Section 3.1.2.1), checking guidelines and requirements (Section 3.1.2.2), detecting damage and tampering (Section 3.1.2.3) and finally, document analysis (Section 3.1.2.4). We present a full overview of the literature in Table 3.2.

3.1.2.1 Completeness and Occupancy

Checking for completeness, e.g. by counting the number of goods present, or retrieving the occupancy status of transportation containers and areas can play an important role to improve and speed up processes in logistics.

Li et al. [Li+12] tackle the problem of monitoring warehouse order picking. They utilize a MS Kinect to detect the picked items and check if any picking errors occur. They restrict themselves to static box-shaped objects placed in a static basket and use 2D texture information as well as 3D geometric information to

match recognized items to a database. More precisely, they use SCARF [TMS11] descriptors and combine them with a volume estimation. The evaluation of the approach yields a recognition accuracy close to $100\,\%$ for most of the eight objects that were tested in this limited scenario.

Özgür et al. [ÖAN16] compare two approaches to recognize the occupancy status of the load handling device of forklift trucks. One approach is sensor-based where an ultrasonic distance sensor is mounted onto the fork mast. A pallet is recognized by monitoring the measured distance. The second approach is camera-based, where the camera is mounted onto the ceiling to have a physically stable environment. Fiducial markers are used to recognize the forklift and a color pattern is applied to the fork. Finally, training data is gathered and a Support Vector Machine (SVM) is trained. The authors do not present quantitative results, however, mention that the sensor-based approach is superior since the configuration effort and the cost for installation and maintenance are lower while the accuracy is higher.

Dörr et al. [Dör+20b] develop a system for automated packaging structure recognition, where the goal is the localization of uniformly packed pallets and the analysis of their composition. They use a multi-step process: pallets are detected and for each pallet, the side faces are segmented. The side face segmentation is rectified using a projective transformation. On the rectified side face, a CNN is used to count the number of parcels and finally, the full packaging structure is determined. The training and evaluation dataset contains a total of 1267 images. The inter-unit segmentation, i.e. the segmentation of the pallets is reported with 0.9943 precision and a recall of 1. The mean image error that takes all instances in the image into account is 0.1564.

In follow-up work, Dörr et al. [Dör+21] present a novel approach for the side face detection problem. They extend CornerNet [LD18] to support the detection of arbitrary four-cornered polygons, instead of axis-aligned bounding boxes. Their new model TetraPackNet shows significant improvements over a Mask R-CNN on the dataset presented in [Dör+20b]. More precisely, the Mask AP increases from 58.7 to 75.5.

Finally, Li et al. [Li+21] present an approach to recognize congestions on conveyor belts. They use pre-processing steps in order to normalize the image of the observed area and subsequently employ edge detection techniques. They separate moving edges from static edges and use statistical information on static edges to make a prediction. The evaluation is performed on $160,000$ videos that were manually labeled, and they report Area Under the Curve of the Receiver Operating Characteristic (ROC-AUC) of 0.9999, which outperforms deep learning baselines they compare against.

Summary Occupancy has been analyzed for forklift masts [ÖAN16] and conveyor belts [Li+21]. Completeness checks have been investigated for pallets [Dör+20b; Dör+21] and order picking [Li+12].

Outlook As the applicable computer vision techniques resemble those from Section 3.1.1.2, we refer to the respective section. Further applications include monitoring and verification of the complete packaging process of a pallet or truck. By following the whole procedure, information is processed sequentially, which alleviates the common issue of occlusion. Also, counting the number of pallets within a truck or in a loading area is a relevant application.

3.1.2.2 Guidelines and Requirements

There are several guidelines and requirements when transporting dangerous goods (cf. [Uni19]). These guidelines help to protect the personnel handling the goods and the freight itself if they are recognized and followed carefully.

Brylka et al. [BBS21] work on identifying dangerous goods by detecting the respective labels on parcels. They generate a dataset of 1,000 manually labeled images and 50,000 synthetically generated images, which can be used for barcode and label detection. The evaluation for barcodes is performed on a dataset, which has over 400 images with 840 barcode instances. They improve upon Brylka et al. [BSB20] by $5\,\%$ in recall. For the detection of dangerous goods labels a

new validation dataset is created by manually labeling 2,260 images with a total of 5,820 labels. Since they consider passing under a camera arch, which yields a sequence of images, they report the detection rate per sequence. The highest observed detection rate per sequence is $96.2\,\%$ at a recall of 0.385 and a precision of 0.976. Note, that they also tackle volume estimation. The respective approach and results are reported in Section 3.1.1.4.

Summary Literature on verifying guidelines and requirements is very limited and focuses on classifying dangerous goods [BBS21].

Outlook In addition to dangerous goods labels, also transportation requirements regarding orientation and maximum load can be investigated. The former refers to checking whether a package can be transported upside down, while the latter refers to labels regarding sensible goods such as glass. Furthermore, transportation units such as pallets might have special packaging requirements (e.g. regarding the lid, foil or straps), that can be verified automatically.

3.1.2.3 Damage and Tempering

Logistics goods can be damaged or tampered with at any point in the supply chain and due to the steadily increasing presence of valuable goods such security considerations gain importance [NZO18]. In order to pinpoint the time and place where such events occur, it is necessary to be able to recognize them automatically. Damages can have several forms, such as water damage or deformation. Also, tampering can be detectable in different ways: e.g. new tape can be applied after opening a parcel or labels could be attached to or removed from a parcel.

Noceti et al. [NZO18] investigate damage and tampering detection in a postal security framework by extracting 3D shape and appearance information (i.e. brightness patterns) from multiple cameras. The authors present their detection method along with use-cases and a database storage of collected data for future reference, however, we will focus our attention on the vision-based detection approach. Change

detection [Sta+15] is used to fit parallelepipeds on binary masks of parcels. Damage detection is then performed by comparing the obtained 3D shape with its expected shape, while tampering is based on the comparison of the parcel's side surfaces [DT05]. Noceti et al. [NZO18] report, that they reach an overall damage and tampering detection accuracy of over 90%.

Malyshev et al. [Mal+21] present a concept outlining the use of CNNs for damage detection. They name deformation, rupture and moisture as possible categories of damage to a package. Also examples of damages, which do not imply damage to the cargo are visualized exemplarily.

Naumann et al. [Nau+23a] present a novel architecture for 3D shape reconstruction of cuboid-shaped and damaged parcels from single RGB images. They combine estimating a 3D bounding box [Bra+23] with an iterative mesh refinement branch [Wan+18], to leverage the strong prior in form of the 3D bounding box while at the same time being able to adjust to damaged parcels. Thus, their approach estimates the 3D mesh of the current, potentially deformed parcel shape, as well as its original pristine shape. This enables not only damage classification but also damage quantification by directly comparing 3D meshes. Training and evaluation are performed on Parcel3D, a novel synthetic dataset of intact and damaged parcels in diverse environments. Their architecture CubeRefine R-CNN performs best on intact parcels (Mesh AP_{50} of 92.8) and competitively for damaged ones (Mesh AP_{50} of 70.7).

Summary Literature on damage and tampering detection is scarce and focuses on parcels [NZO18; Nau+23a]. The most important requirement for such systems, according to the results of the questionnaire by Noceti et al. [NZO18], is easy integration into existing processes. More precisely, current processes should not be slowed down significantly and hardware installation should be easily possible within existing plants. This, however, is challenging since the process and plant design can differ significantly within and across companies.

Outlook Tampering detection approaches rely on generating viewpoint invariant parcel side surface representations, which can be computed from the parcel corner points. Thus, incorporating recent advances in keypoint detection seems promising. Moreover, prior knowledge of the cuboid shape can be leveraged by utilizing a vanishing point loss [Rui+20] or exploiting 2D/3D correspondences [Li+20]. Damage pattern recognition, i.e. identifying and clustering damages to recognize frequently occurring patterns is very interesting. In addition to that, estimating the full 3D reconstruction of a parcel from RGB-D data has not been studied yet and would be very interesting to investigate for enabling detailed damage quantification. Finally, there is no dataset for analyzing different cases of damages, such as water damage or ruptures in the packaging, and damages for other objects such as pallets have not been investigated yet.

3.1.2.4 Document Analysis

Shipments are always accompanied by documents that provide additional insights such as product types, product quantities, and product prices. Thus, to obtain detailed information about the shipment, it is also necessary to automatically process documents. The first step towards this goal is to unwarp and rectify documents that might have been crumbled or warped during the transportation process. This task is called document rectification and it has gained a lot of attention recently [Ma+18; Xie+20; Mar+20; Fen+21; Xue+22; Das+22; Jia+22; Wan+22]. Once the document is rectified, contents can be analyzed using OCR. We refer to reviews for handwritten [Mem+20] and typed [Sub+21] OCR. Moreover, document structure recognition plays an important role [Pin+03; Ria+17; Sub+21; Li+22].

Summary Approaches in the area of document analysis tackle the general problem and we are not aware of literature focusing on logistics use-cases. Document rectification [Ma+18; Xie+20; Mar+20; Fen+21; Xue+22; Das+22; Jia+22; Wan+22] has gained a lot of attention recently. Moreover, OCR [Mem+20; Sub+21] and document structure recognition [Pin+03; Ria+17; Sub+21; Li+22] are important problems that are investigated in the literature.

Outlook The above-mentioned techniques can be applied to documents that are relevant to logistics processes, such as delivery notes. Especially detecting and interpreting human annotations on such documents seems very promising.

3.2 Manipulation" [Nau+23b]

While for monitoring the focus was on information retrieval, we now focus on the interaction with the environment. We follow Borstell [Bor21] and divide this section according to the degree of automation into two parts: machine-supported tasks and fully autonomous manipulation. The former, also called assistance, refers to providing helpful additional information to human operators and will be treated in Section 3.2.1 and is summarized in Table 3.3. The final objective is not to fully automate a process, but instead to ease the workload for operators. Literature on fully autonomous manipulation will be presented in Section 3.2.2 and is summarized in Table 3.4. In contrast to the literature on assistance, the focus here is not on providing additional information but on helping to solve the task at hand autonomously. Note, that literature is categorized w.r.t. the goal that is pursued and not the achieved degree of automation.

3.2.1 Assistance for Manual Manipulation

Nowadays, most logistics processes are still handled manually. To ease the workload for human operators, assistance systems can be developed. Note, that the research presented here specifically strives towards assisting a human operator, as opposed to aiming at fully automating a process. We identified literature focusing on order picking, which is discussed in Section 3.2.1.1 and on the packaging process, which is presented in Section 3.2.1.2. For a general overview of all literature, we refer to Table 3.3.

3.2.1.1 Order Picking

Orders commonly comprise several items, which need to be collected for shipping. The process of assembling an order is also referred to as order picking, and it is essential for the efficiency of warehouses.

Reif [Rei09] investigates the suitability of Augmented Reality (AR) for order picking from an operator's perspective. During order picking, static text with product information as well as dynamic 3D information, e.g. , about the source or target position, can be displayed. The authors perform a study that shows a steep learning curve and an improvement in order picking time and error rate in comparison to order picking with a paper-based list. Further, the subjective workload is not higher when using AR, while the mental workload is lower due to the provided helpful information. Nonetheless, they recommend further longtime studies to analyze the effects and note that the learning curve depends on the individuals.

Grzeszick et al. [Grz+16] present an approach to assist the picking process with wearables. The picker is equipped with a smartwatch and a low-cost camera. The camera is used for activity recognition, which triggers further processing if determined that the current activity is picking an object. When the activity recognition recognizes a picking process an image is taken and analyzed with barcode detection and a CNN to check whether the correct item was picked. The smartwatch is used to display the information relevant to the next pick and gives tactile feedback regarding the success of the pick. The authors report 80.1% accuracy for action recognition and 89% for the recognition of clearly visible barcodes.

Summary Assistance solutions for order picking focus on using AR [Rei09] and wearables [Grz+16]. In both cases, results suggest a steep learning curve and a positive impact on the operator's performance.

Table 3.3: Overview of the literature on Assistance for Manipulation (cf. Section 3.2.1)

Paper	Summary	Application	Data Type	Public Dataset	Approach Type	Objects
[Rei09]	Analysis of the suitability of augmented reality for order picking	Order Picking				
[Li+12]	Monitoring warehouse order picking	Order Picking	RGB-D, Real		Classical Approach	Other Packaging
[Grz+16]	Assist the picking process with wearables	Order Picking	Real		Deep Learning	Label
[Hoc+16]	Assistance system based on augmented reality for quality control during the packing process	Packaging for Shipment	RGB, RGB-D, Real			Other, Parcel
[Mät+16]	Study to analyze how the packaging process can be improved by using augmented reality	Packaging for Shipment	Real			

Table 3.4: Overview of the literature on Autonomous Manipulation (cf. Section 3.2.2)

Paper	Summary	Application	Data Type	Public Dataset	Approach Type	Objects
[Tha+13]	Pointcloud segmentation for container unloading	Depalletization	Pointcloud, Real, Synthetic		Classical Approach, Pattern Matching	Container/ Trailer
[Tha+14]	Pointcloud segmentation for container unloading	Depalletization	Pointcloud, Real, Synthetic		Classical Approach, Deep Learning, Pattern Matching	Container/ Trailer
[Pra+15]	Depalletization using a robot arm and low-cost 3D sensors	Depalletization	RGB-D, Real, Synthetic		Classical Approach, Pattern Matching	Pallet, Parcel
[Arp+20]	Optimized depalletization using a single RGB-D camera	Depalletization	RGB-D, Real		Classical Approach	Pallet, Parcel
[Chi+20]	Depalletizing using a robot with a fixed time-of-flight camera and an eye-in-hand RGB camera	Depalletization	RGB, RGB-D, Real		Classical Approach, Pattern Matching	Parcel
[BBS21]	Detection of dangerous goods labels and volume estimation by pointcloud segmentation	Depalletization, Label Detection, Volume Estimation	RGB, RGB-D, Real, Synthetic		Deep Learning	Label, Pallet, Parcel
[VN14]	Pallet detection and localization for an autonomous forklift using a stereo camera	Pallet Handling	RGB, Real		Classical Approach, Pattern Matching	Pallet
[VCN15]	Pallet detection and localization for an autonomous forklift using a stereo camera	Pallet Handling	RGB, Real		Classical Approach, Pattern Matching	Pallet
[HBC16]	Pallet engagement in the context of military logistics	Pallet Handling	RGB, RGB-D, Real		Classical Approach, Fiducial Markers	Pallet

Table 3.4 (continuation): Overview of the literature on Autonomous Manipulation (cf. Section 3.2.2).

Paper	Summary	Application	Data Type	Public Dataset	Approach Type	Objects
[VN16]	Pallet detection and localization for an autonomous forklift using a stereo camera	Pallet Handling	RGB, Real		Classical Approach, Pattern Matching	Pallet
[Xia+17]	Pallet recognition and localization by using an RGB-D camera	Pallet Handling	RGB-D, Real		Classical Approach	Pallet
[MF18]	Pallet detection and localization using a time-of-flight camera	Pallet Handling	RGB-D, Real		Classical Approach, Pattern Matching	Pallet
[MF19]	Driver assistance system for a forklift truck	Pallet Handling	RGB-D, Real		Classical Approach, Pattern Matching	Pallet
[Moh+20]	Pallet recognition and tracking using only an onboard laser rangefinder	Pallet Handling	Pointcloud, Real		Deep Learning	Pallet

Outlook In general, AR has a high potential for assistance during manual manipulation processes. Its usefulness for the area of logistics has been analyzed by Stoltz et al. [Sto+17] and an industry perspective is provided by DHL [Glo+20]. Recent plane detection algorithms [Liu+19] can help to reduce sensor requirements to enable a broader acceptance of AR techniques. Furthermore, it would be interesting to investigate the suitability of AR interfaces for different applications, since for example glasses or a projector might not be feasible for some use-cases. Finally, research on the utility of wearables in different logistics scenarios is an interesting topic for research.

3.2.1.2 Packaging for Shipment

Once all items of an order have been assembled, they need to be stored safely and efficiently inside a transport unit, such as a parcel. This process is crucial for customer satisfaction since damaged items are a frequent cause for complaints.

Hochstein et al. [Hoc+16] develop an assistance system based on AR for quality control during the packing process. They use one RGB camera and one time-of-flight camera (ToF camera). Commercial software for object detection is used for monitoring the packing process. Hochstein et al. [Hoc+16] do not recognize all articles that are placed into a parcel but focus on the localization within the box. As assistance, the packing list and other relevant information are projected onto the workplace. The authors constructed a working prototype focusing on ergonomics and privacy, however, no user study was conducted.

Mättig et al. [Mät+16] perform a study to analyze how the packaging process can be improved by using AR. They name pressure on time, quality, and costs as potential areas of improvement enabled by the usage of AR for packaging. They perform a study with two groups of 10 people and the results confirm the hypothesis that AR helps to improve time efficiency when packing order suggestions are displayed. In addition, it helps optimizing costs, i.e. identifying the best parcel size.

Summary Literature on assistance during the packaging process focuses on the usage of AR techniques [Hoc+16; Mät+16]. Overlaying digital information with the visual perception of the world can help to improve efficiency and reduce errors during the packaging process.

Outlook Since the same techniques as mentioned in Section 3.2.1.1 are relevant, we refer to the respective outlook. In addition to that, further in-depth studies of the performance and acceptance of such techniques are promising research directions.

3.2.2 Autonomous Manipulation

While most tasks are still handed by humans or by Human-Computer-Interaction (HCI), research is striving for fully autonomous solutions. In this section, we present research that works towards this goal. Since we prioritize the long-term objective of automation, works are presented independent of the current level of automation that they achieve. First, literature on pallet handling is reviewed in Section 3.2.2.1, and subsequently literature on depalletization in Section 3.2.2.2. Afterwards, we present insights into logistics-related approaches for pick-and-place in Section 3.2.2.3 and automated guided vehicles (AGVs) in Section 3.2.2.4 briefly. Moreover, we provide a general overview of the literature in Table 3.4.

3.2.2.1 Pallet Handling

Pallet handling refers to the task of automatically recognizing, localizing and interacting with pallets. Due to the ubiquity of pallets in the context of material handling, this is a crucial task that is frequently needed in logistics processes.

Varga et al. [VN14] present an approach for pallet detection and localization for AGVs using a stereo camera. During the considered loading and unloading operations, the AGV is assumed to stop at a distance of approximately $2.5\,\mathrm{m}$

from the pallet. To engage with the pallet, the 3D position must be provided accurately. Accuracy is defined as a deviation of at most $5\,\mathrm{cm}$ and $1°$. The presented pipeline consists of first performing stereo image rectification and stereo matching. Subsequently, pallet detection is applied to the left image. This result is then employed for exterior reconstruction and plane fitting. They define a model for a pallet, which consists of three legs that are separated by empty pockets. The relative proportions of each area are assumed to be known and constant. They use a sliding window approach and design features that represent the assumed pallet model. These features are used to train an AdaBoost [FS95] classifier. The datasets for training and testing were gathered manually for this work. The detection rate for weak positives is reported as $94\,\%$, the one for strong positives as $84\,\%$, while having a false positive rate of $1.5\,\%$.

Varga et al. [VCN15] follow up on their work [VN14] and again use a stereo camera system to detect pallets for an autonomous forklift. Their approach is based on a fixed-size sliding window, where multiple scales of the image are used as input and a fixed aspect ratio of the pallet is assumed. They compute eight image channels based on the camera input: a grayscale image, the gradient magnitude and the oriented gradient magnitude at six different orientations. A random forest (Adaboost [FS95]) is used to train a classifier. The training dataset Viano2 contains 7,124 images with 9,047 pallets and the test dataset Viano3-5 contains 467 images with 891 pallets. Their best model achieves $78.1\,\%$ precise matches on the test set. The authors also perform a test, during which all operations were successful after fine-tuning for the new scenario. They name illumination as an open challenge, which is tackled in their subsequent work [VN16].

Haanpaa et al. [HBC16] focus on the problem of pallet engagement in the context of military logistics. A multi-sensor setup is used that comprises two ToF cameras and an RGB camera. They consider a pipeline consisting of different steps and vary the type of input information. In addition to that, they resort to fiduciary markers for certain tasks. The authors do not provide a quantitative analysis of their results.

Xiao et al. [Xia+17] present an approach for pallet recognition and localization by using an RGB-D camera. A region-growing algorithm [Xia+13] is used to segment the depth image into planar patches. A heuristic for pre-processing is employed, and afterwards, pattern matching on the remaining segments is applied. The pattern matching focuses on the pallet only and a set of five different pallet base models is used. The pattern matching does not rely on any color information but instead works on a binary image that belongs to a planar patch. The authors present qualitative data to exemplary show how well the approach works, however, no quantitative results are presented. As failure cases, they mention that the pallet could be too dark or that items obscuring the pallet can change the observed pallet pattern.

Molter et al. [MF18] present an approach for pallet detection and localization using a ToF camera. The camera is mounted on top of the back side of the fork of a forklift. They remove the ground plane within the pointcloud using a RANSAC [FB87] approach. Subsequently, a region-growing algorithm is used to determine surface clusters. For each cluster, the centroid and normal vector are calculated, and filtering for vertical planes is applied. Afterwards, centroid pairs and triple pairs are computed and finally, a check using the geometrical information on the pallet is performed to obtain a candidate. The authors report, that in the static scenario, almost all pallets were detected correctly, while in the dynamic one only $55\,\%$ were localized correctly. In a follow-up work, Molter et al. [MF19] present an advanced driver assistance system for forklifts operated by humans. The driver assistance system utilizes the pallet detection approach [MF18] and combines it with trajectory planning and control for the forklift, as well as a user interface inside the forklift.

Mohamed et al. [Moh+20] present an approach for pallet recognition and tracking using only the onboard laser rangefinder of a forklift. The data from the laser scanner is used to create top views of the occupancy of the floor. These top-view images are then processed by a Faster R-CNN [Ren+17]. To enable robust localization and tracking over time a Kalman filter [Kal60] is used. The authors evaluate on a real-world dataset containing 340 labeled top-view images, which is augmented by rotation and displacement to 1020 images. 714 samples are

used for training and 306 are used for testing. An average accuracy of 99.58 % is reported.

Summary The problem of pallet localization has been tackled frequently in literature. Most approaches employ classical computer vision techniques and frequently pattern matching is part of the pipeline [VN14; VCN15; VN16; HBC16; Xia+17; MF18]. Recently, deep learning has been used to identify pallets from laser rangefinder data [Moh+20]. For details on earlier approaches, we refer to the literature review on the pallet loading problem by Vargas-Osorio et al. [VZ16] and to Mohamed et al. [Moh+20].

Outlook For pallet engagement, the usage of recent pointcloud classification and segmentation algorithms (cf [GMR17; Bel+20]) has not yet been investigated. Due to the high accuracy of state-of-the-art 2D object detection algorithms [Xie+21] they are very well-suited for pallet detection, provided a sufficiently large dataset is available.

3.2.2.2 Depalletization

In warehouses, hand-held scanners are still very common to verify that a package is on the correct pallet and to do inventory. Also the subsequent step of depalletization involves a lot of human labor and can still be slow and error-prone.

Thamer et al. [Tha+13] develop a segmentation technique for pointclouds in logistics applications. More precisely, they present a system for the automatic unloading of containers. With the knowledge of the spatial relationship between the sensor and the container, they filter out the background in a first step. The pointcloud now only contains objects of interest and needs to be segmented. They pursue a two-step approach inspired by [Sch+07; STU11] and start with applying the graph-based segmentation technique by Felzenszwalb et al. [FH04]. To improve on these results, they additionally implement an iterative process for region-growing. Subsequently, Thamer et al. [Tha+13] fit surface patches onto the

segments of the pointcloud [Mor78]. In order to find objects within the surface patches, they define models for boxes, cylinders and sacks which they try to fit. The fitting process consists of a directed graph, where the surface patches are the vertices and the edges save information on possible object matches. By employing distance metrics, they find possible candidates for objects and check them against the models for their predefined shapes. The evaluation of the approach is done with 54 different real-world packaging scenarios and further 51 synthetically generated ones. They report better performance on the synthetic data, recognizing $83\,\%$ of the labeled graspable goods within the scene. The authors argue, that the approach has high potential, however, was not capable to run in real-time in 2013.

Thamer et al. [Tha+14] is a subsequent work, which also tackles the problem of detecting differently shaped logistics goods to enable automated processing by robotic systems. The classes box, barrel and sack are considered and artificial training data is generated for each class. Their approach does not operate on the points directly, but instead uses Viewpoint Feature Histograms (VFH) [Rus+10]. As pre-processing steps, points belonging to the background are removed and denoising techniques are applied. In addition, object candidates are separated by applying clustering on their Euclidean distance. SVMs and ANNs are used for the final classification and trained on 500 simulated training examples per shape class. The authors report a classification accuracy above $90\,\%$ on synthetic data, while the performance on real data is significantly lower in most cases. Moreover, SVMs seem to perform better on synthetic data, while the ANN seems more robust and handles real data better.

In 2015, Prasse et al. [Pra+15] used a robot arm and low-cost 3D sensors to perform the task of depalletization. They present two approaches: The first uses a Photonic Mixing Device (PMD) sensor and a pre-determined model of loading situations, while the second employs a 3D scanning approach by dynamically positioning a structured light sensor with the robot arm. The loading units are assumed to have a cuboid shape and are determined by employing a Random Sampling Consensus (RANSAC) [FB87] algorithm. Prasse et al. [Pra+15] evaluate the PMD approach on real data and report the deviation of package dimensions for 9 parcels. Since the deviation in height is less than their threshold of $10\,\mathrm{mm}$, they argue that the

approach is suitable for application in logistics. The second approach is evaluated on synthetic data only and the influence of the following factors on accuracy and runtime are investigated: pointcloud size, number of iterations, parcel dimensions, randomly transformed pointclouds, simulated Gaussian sensor noise, and number of detected parcel faces.

Arpenti et al. [Arp+20] present an approach for depalletization using a single RGB-D camera. The information from the RGB-D camera is supplemented by a case database, which contains the product barcode, the number of boxes on a pallet, their dimensions and one image of each textured face of the box. Since they have an existing database, image segmentation is based on matching the SIFT [Low99] features of the textures. If sufficient matches are found, the homography between the taken image and the one from the database is computed. Because this only works for textured faces, the watershed transform [Mey92] is used for the other cases. The segmentation is then used, in order to determine the 3D plane of the face from the depth information and all points from the segmentation are associated with it. The minimum area enclosing rectangle around this pointcloud is then used as input for the geometrical module. The purpose of the geometrical module is to find matching faces for the candidate faces in the database. This is done by matching the visible dimensions to the database information and the pose is estimated. For the evaluation, a database of nine cases that are organized in ten different settings is considered. The authors report an accuracy of $98\,\%$. The authors also perform a case study on the whole depalletizing process. Here, a black tendon is placed behind the pallet to reduce reflections for the RGB-D camera. One configuration was tested and all parcels from this configuration were correctly depalletized.

Chiaravalli et al. [Chi+20] present an approach towards depalletizing using a robot with a fixed ToF camera and an eye-in-hand RGB camera. The sizes of the boxes as well as the plane through the top of the highest parcel are assumed to be known. Canny edge detection [Can86] and the Hough transform [Hou62] are used to determine the edges of the boxes. A connectivity graph is then used to generate box hypotheses and an optimization problem is solved using a genetic algorithm to identify the relevant boxes. Afterwards, their pipeline comprises the following

steps: gap localization, gap alignment, insertion test, insertion complete, and collection. Note, that the goal is not to grasp the parcel, but instead to pull it to a desired position. The box detection and pose estimation is analyzed on 125 depth images. The authors report an average standard deviation of $3.05\,\mathrm{mm}$ for the box position and a position error of $3.60\,\mathrm{mm}$. For evaluation, six boxes of equal size are positioned in three rows of two parcels each, with no initial gap.

Summary All approaches use depth information for depalletization and rely on base patterns. Most approaches operate on pointclouds [Tha+13; Pra+15; Chi+20], while [Tha+14] rely on Viewpoint Feature Histograms (VFH) [Rus+10]. Currently, approaches do not use GNNs for this task and oftentimes additional prior information is assumed.

Outlook Similar to the case of pallet handling, the usage of pointcloud classification and segmentation algorithms (cf. [GMR17; Bel+20]) has not yet been investigated and suggests interesting future applications. Furthermore, the approaches work towards automated pallet handling, however, to enable reliable real-world deployment higher robustness and end-to-end integration are necessary.

3.2.2.3 Pick-and-Place

The importance of pick-and-place for logistics use-cases is manifested for example in the Amazon picking challenge [Cor+18]. Since the applications of this task reach far beyond logistics, we refer to existing literature reviews [BK00; Bai+20; Kle+20; Du+21]. Works with a focus on logistics include [Ren+16; Zen+17; BUE16; Sch+17; Wah+19; Pav+19].

3.2.2.4 AGVs

AGVs are important in the logistics domain and beyond [CCW19; Fot+21]. Key components of systems include visual SLAM [MHH19] and planning & control [Fra+21]. Research with a particular focus on logistics includes [HM17; KTS18; Sab+18; YS19; Zho+21]. Note, that also autonomous driving has a huge potential for logistics [MV19], however, is not considered as part of this review.

3.3 Computer Vision Perspective" [Nau+23b]

The goal of this section is to review the presented literature from a computer vision perspective. We first present a categorization of the literature w.r.t. the computer vision tasks they tackle in Section 3.3.1. Subsequently, we present a brief overview of existing, publicly available datasets in Section 3.3.2 and of industrial solutions in Section 3.3.3.

3.3.1 Methodological Categorization

We briefly categorize the reviewed literature according to the computer vision task they solve. Approaches for 2D and 3D data are presented, and we refer to Riestock et al. [Rie+19] for an overview of sensors commonly used in logistics.

Marker-based Detection: Fiducial markers are commonly used to facilitate visual identification. Those markers include barcodes [Mis+19], ArUcos [HBC16] and other markers [Gar+98; Rei09; Wei+10]. Using markers is still a valid and robust approach, where the overhead of marking all necessary goods is feasible.

Edge Detection: Edge detection has been applied for interacting with pallets [MF18] and parcels [Nau+20].

Object Detection: Several approaches [VN14; VN16; She+12; May+20; Mal+21] tackle the problem of object detection, i.e. localizing and classifying an object in

an image. Apart from RGB images, object detection has been applied to top-view occupancy images of 2D laser range finders [Moh+20].

Instance Segmentation: Instance segmentation tackles the problem of identifying each instance of a class separately by assigning pixel-wise correspondences. Research in the area of logistics includes [ST14; Suh+19; Nau+20; MGF20; BBS21; Li+21], where packing structure recognition [Dör+20b; Dör+21] exploits this information to determine the pallet composition. Furthermore, [HM17; Sun+20] use instance segmentation together with depth information.

Object Re-Identification and Tracking: The same object might occur multiple times in images, e.g. at a different location for a different point in time. The goal of object re-identification is to be able to find all images in which a target object is visible. A very popular area of application for this task is person re-identification. The existing literature considers boxes [Li+12], parcels [NZO18], and pallets [Rut+21; Rut+22a; Klü+22]. Moreover, work on tracking parcels [Cla+19] has been proposed.

Action Recognition: Identifying the activity currently pursued by a person can be a very helpful task. For example, research has been done on determining if a worker is currently picking an object, to use this information for order picking [Grz+16].

3D Object Detection: Plane segmentation is used for detecting parcels [Hu+21a] and pallets [Xia+17; MF18]. Moreover, clustering approaches [FZL21] and deep learning-based pointcloud classification [BBS21] have been used.

3D Shape Reconstruction: There are approaches for dataset generation that heavily rely on fiducial markers [Mih+15]. Furthermore, digital measurement with RGB-D cameras has been investigated in static scenes with traditional methods [Küc13; Son+17a; Kuc+19]. Finally, Naumann et al. [Nau+23a] tackle single image 3D reconstruction for assessing parcel damages.

3.3.2 Datasets

The availability of freely available datasets is limited, as we can infer from Tables 3.1, 3.2, 3.3 and 3.4. Datasets exist for instance segmentation of logistics objects [May+20] and parcels [Nau+20; Nau+23a]. Furthermore, chipwood re-identification [Rut+21; Rut+22a] datasets have been presented. Most of the presented works, however, utilize datasets that are not available to the public and are mostly described only briefly.

3.3.3 Industry Services

Due to the huge potential of computer vision applications in logistics and the market potential of such solutions, numerous companies offer commercial products and services in this area. We present an extract of relevant companies and selected products in Table 3.5. This overview shows that companies are actively working on solutions for all the areas mentioned in Section 3.1 and 3.2.

The overview presented here is not complete. We provide further details online on our project website and invite the audience to use it and contribute.

3.4 Discussion

We presented a broad overview of computer vision applications in transportation logistics and warehousing, targeted to introduce a diverse audience to the topic. Since this work, however, focuses on a subset of the presented applications, we briefly summarize the findings of the literature review w.r.t. the objectives of this work.

Table 3.5: Overview of Industry Solutions (cf. Section 3.3.3).

Company	Product	Description	Application
CARGOMETER GmbH	CARGOMETER	On the fly measurement and weighing for a moving forklift. Hardware is installed at loading gates, where every pallet has to pass by.	Object Detection, Pallet Handling, Tracking and Tracing, Volume Estimation
Cognex	3D-A1000 dimensioning system	Capture moving objects in 2D and 3D for identification, volume estimation and damage detection.	Damage and Tampering Detection, Label Detection, Object Detection, Volume Estimation
FIZYR	Parcel handling	Flexible AI-powered software solution, e.g. for parcel pick-and-place. Detects deformation, damages, wrinkles and labels.	Label Detection, Object Detection, Pick-and-Place
HIKROBOT	Single Piece Separation System	Real-time positioning of packages to ensure that packages pass through a gate individually and at a pre-defined rate.	Sorting
KEYENCE	AGV safety precautions	Laser scanner for AGVs to ensure safety	AGVs, Object Detection
LMI	3D Scanning and Inspection for the Packaging Industry	Package filling inspection by scanning and 3D measuring to ensure the correct fill-level	Object Detection, Quality Control
Loadscan	Load Volume Scanner system	Measuring the volume of the material loaded in a truck or trailer bin. Based on laser scanner and proprietary software.	Volume Estimation
Locus Robotics	Directed Picking	Support during the picking process by actively directing workers to their next picking location.	Order Picking
Omron	Vision and Inspection	Label inspection for ISO and GS1 standards	Label Detection
River Systems	Mobile Sort	Picking and sorting solution to enable batch picking with subsequent sorting into distinct orders.	Order Picking, Sorting
TeamViewer	xPick	Augmented reality solution to support order picking, sorting, inventory control and other processes in logistics and warehousing	Augmented Reality Assistance, Order Picking

Table 3.5 (continuation): Overview of Industry Solutions (cf. Section 3.3.3).

Company	Product	Description	Application
VC Vision Components	Advanced Driver-Assistance System	Advanced driver-assistance systems for forklifts	AGVs
VITRONIC	Air Freight Measurement	Air Freight Measurement system	Volume Estimation
Valerann	Monitor	Monitoring of road traffic	Vehicle Traffic
Zebra	Packing and Staging	Order confirmation and shipping label printing for hands-free packing and staging in warehouses and distribution centers.	Other Assistance

One goal of this thesis is to develop a robust parcel segmentation approach. Naumann et al. [Nau+20] present an approach for robust parcel segmentation without the necessity for task-specific training data. However, since their approach does not rely on a unified backbone, it cannot be leveraged for downstream tasks. In contrast to that, Clausen et al. [Cla+19] use manually annotated training data to fine-tune a backbone that could be utilized for downstream tracking applications such as keypoint detection or 3D reconstruction. Due to the reliance on manual data annotation, their approach is not able to efficiently cope with potentially occurring domain shifts. Thus, we tackle this open problem in Chapter 4.

Tampering detection for parcels has previously been tackled by Noceti et al. [NZO18]. However, the authors use a multisensory setup and prescribe a predetermined environment. This significantly reduces the complexity of the task and renders their approach inapplicable for scenarios such as last-mile delivery. We work towards removing these constraints to enable flexible tampering detection based on a single RGB image only in Chapter 5.

Finally, while volume estimation for pristine parcels using RGB-D cameras has been investigated [Kuc+19; BBS21], Noceti et al. [NZO18] are the only ones considering damage detection. Again, their approach relies on a multisensory setup and an a-priori known environment. We strive to remove these restrictions by presenting an approach for single image damage assessment in Chapter 6.

4 Robust Parcel Segmentation

Reliably identifying the object of interest in an image is crucial for downstream tasks such as keypoint detection or 3D reconstruction. However, ready-to-use datasets of sufficient size are available only for a limited number of domains in practice. Since the generation of real-world datasets is time-consuming and expensive, synthetic datasets have gained increasing attention [Nik21]. In this chapter, we extend the simple, yet effective dataset generation approach by Dwibedi et al. [DMH17]. The original work combines pasting objects of interest and distractors onto random backgrounds with the application of different blending methods to prevent Artificial Neural Networks (ANNs) from overfitting to potentially unrealistic and characteristic margins around the pasted object — the so-called *local pasting artifacts*. By adding automated image scraping and selection, we extend their approach to cover the entire dataset generation pipeline as summarized in Fig. 4.1. Due to our approach's simplicity, it is suitable for dataset creation and updates, e.g. when a domain shift occurs. A case study on parcel detection and segmentation is carried out to evaluate our approach. To summarize, the main contributions in this chapter are:

- we extend the work of Dwibedi et al. [DMH17] by adding image scraping and image selection, which results in a complete and easily usable pipeline for dataset generation for instance segmentation,
- we investigate the influence of image selection and blending methods on the performance for transfer learning,
- we introduce a novel real-world image dataset of parcels with 2D and 3D annotations, and

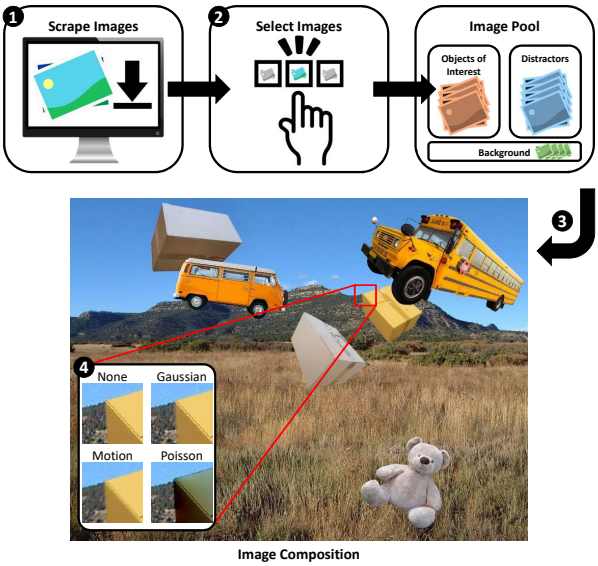


Figure 4.1: Overview of the dataset generation pipeline: (1) We scrape relevant images from image search engines. (2) These images are filtered by applying three different image selection methods: basic pre-processing, manual selection and CNN-based selection. (3) We use the resulting image pool to generate novel image compositions by pasting selected objects onto random background images. (4) Four different blending methods are applied to enable invariance to local pasting artifacts. [Graphic from [Nau+22], ©2022 IEEE]

- finally, our code is publicly available at https://a-nau.github.io/parcel2d to facilitate generating instance segmentation datasets for the community.

The remainder of this chapter is organized as follows. Related literature is presented in Section 4.1. Subsequently, we introduce our dataset generation approach in Section 4.2 and evaluate its performance in Section 4.3.

Sections 4.1 to 4.3 have been previously published and are direct quotes from Naumann et al. [Nau+22], including tables and figures. These sections are marked with ❞ [Nau+22] in the respective headline. For the evaluation in Section 4.3.2, we reran our experiments to conduct five different training runs per scenario

to increase the statistical meaningfulness. We adapted Table 4.1 to report the updated results.

4.1 Related Work" [Nau+22]

The idea of generating an artificial training dataset is widespread, due to the high cost that incur for capturing and annotating a tailor-made dataset for a use-case. We first present relevant literature regarding the creation of artificial datasets and subsequently delve into the application area of logistics.

Artificial Dataset Generation. Artificial datasets can either be rendered or composed. When rendering images, we can carefully choose a desired image layout and easily generate a multiplicity of annotations - even the ones that are very costly to obtain, such as 3D annotations. BlenderProc [Den+19] is a procedural Blender[1] pipeline that enables photorealistic renderings to create synthetic datasets. Examples for popular rendered datasets include [Son+17b; Zhe+20].

In contrast to that, image datasets can also be generated by composition. Image composition is the task of seamlessly combining two images by cutting a foreground object from one image and pasting it onto another image. This is an important task in computer vision with a wide range of applications. Niu et al. [Niu+21] present a comprehensive survey on the topic, and we refer to them for details on applications and subtasks included in image composition. For our work, we focus on simple image composition and neglect effects that might make images look unrealistic to humans, as this has proven to be sufficient for training the backbone of a neural network [Ghi+21]. More explicitly, inconsistencies introduced by incompatible colors, unreasonable illumination, mismatching size of objects, or their location are not considered.

[1] See https://www.blender.org/ [Last accessed on Sept. 20, 2024].

Dwibedi et al. [DMH17] present a procedure to generate a targeted dataset for instance segmentation. As input, a set of images for each category, picturing solely the object of interest with a modest background, is needed. They recommend diverse viewpoints, in order to enable detection from diverse viewpoints as well. A foreground background segmentation network is trained to obtain segmentation masks for the foreground objects. In addition, suitable background images need to be chosen. Afterwards, objects are cut out with their mask from the images and pasted onto a background image. Dwibedi et al. ensure invariance to local artifacts from pasting by applying a set of blending methods. The exact same images are synthesized multiple times, where only the blending method varies. They show that this method enables training a neural network for instance segmentation and that combining the synthetic data with only $10\,\%$ of the real training data surpasses the performance compared to training on all real data. Ghiasi et al. [Ghi+21] present a similar technique, however, they use existing annotated datasets as their source for both the foreground and the background and found scale jittering to be very efficient. First two images within a dataset are randomly chosen and their scale is jittered. Subsequently, objects from one image are cut out by using their given annotated mask and pasted randomly onto the second image. During this process annotations within the second image are adjusted accordingly, i.e. adjusted for occlusion. They do not use geometric transformations such as rotation and find Gaussian blurring not to be beneficial. Ghiasi et al. conclude that their method is highly effective and robust. Mensink et al. [Men+21] present a study on the influence of several factors on the performance for transfer learning. They find that the image domain is the most important factor and that the target dataset should be contained in the source dataset to achieve best results.

In our work, we follow an approach similar to Dwibedi et al. [DMH17], however, fully automate the foreground object image retrieval by using web scraping and a pre-processing pipeline.

Applications in Logistics. Work on the plane-wise segmentation of parcels, without the need for a custom training dataset was presented by Naumann et al. [Nau+20]. Plane segmentation information is combined with contour detection

to generate plane-level segmentations. Small load carriers have been targeted using synthetic training data [MGF20]. Furthermore, the problem of packaging structure recognition has been tackled [HM18; Dör+20b; Dör+21]. Packaging structure recognition aims at localizing and counting small load carriers that are stacked onto a pallet.

4.2 Dataset Generation" [Nau+22]

Our dataset generation approach is based on Dwibedi et al. [DMH17]. We follow a similar procedure, apart from the data acquisition approach. This section is organized as follows: In Section 4.2.1, we explain the data acquisition through web scraping. Subsequently, we present three different image selection methods which yield three different datasets in Section 4.2.2. The image generation is explained in Section 4.2.3 and finally we present our real dataset in Section 4.2.4.

4.2.1 Image Scraping

In order to generate a synthetic dataset, it is crucial to have a sufficiently large set of images picturing the object of interest. We approach this problem by scraping images from popular image search engines. We use four different search engines:

- Google Images: images.google.com,
- Bing Images: bing.com/images,
- Yahoo Images: images.search.yahoo.com and
- Baidu Images: image.baidu.com.

We scraped images for the object of interest, i.e. parcels, and for distractor objects. The full source code, including a Dockerized web application is available at https://a-nau.github.io/parcel2d.

Objects of interest. For the objects of interest, i.e. the parcels, we chose 9 different search queries that all represent the same object category: *parcel, parcel package, parcel amazon, packet post, packing carton, packing box, carton box, shipping box and pallet carton*. In order to increase the diversity and quantity of the image data, we translated the English language search queries to German and Chinese. The parcel search was performed in English and German for the search engines Google, Yahoo and Bing. Chinese was used for Baidu. In total, we collected $21,862$ images of the object of interest.

Distractor objects. Since the distractors can be arbitrary objects, we randomly sampled 100 category names from the ShapeNetSem dataset [SCH15] and used these as search query. This gives us a wide range of object categories, while simultaneously preventing the introduction of a strong bias towards certain categories. Since it is easier to find suitable distractors, we only performed the search in English and German and focused on Google Image search. In total we downloaded more than 12,000 images for distractors.

Background Images. We did not scrape background images, but instead used images from the SUN397 database [Xia+16]. We excluded the category *archive* since parcels might be contained in the background, and sampled the scene categories randomly otherwise. Our training, test, and validation split is done across categories, not image instances, in order to prevent a leakage of background image information.

4.2.2 Image Selection

Scraping images by merely a textual input can yield high quantities of data, however, it is difficult to assess the suitability of each of the images for the dedicated use-case. Desirable are images, where

- the image quality is sufficiently high to enable high-quality image compositions,
- the image is a photograph of a real scene,
- the background is homogeneous and easy to remove, and
- the object of interest is the only object and not occluded.

To select such images, we started off by removing all tiny images, i.e. smaller than $80\,\mathrm{kb}$ in size. This threshold was determined empirically, trying to prevent the usage of low quality images. The next step is to analyze the backgrounds of the images. Since we want to cut out objects automatically in the next step, we discard images with inhomogeneous backgrounds. This is achieved by analyzing the color variability of the outer frame of the images. More precisely, we compute the variance of all pixels within a $2\,\%$ outer margin of the image, and keep all images where the mean of the variance of the three color channels is smaller than 50. Subsequently, we apply the automated background removal tool *Rembg*[2] that converts images into masked RGB-A images. The tool is based on *U^2-Net* [Qin+20] and is used to segment the objects of interest. Automated background removal is a challenging task, since the pre-processing only removes images with a strong background variance, however, the images might still contain a cluttered scene with multiple objects. We noticed that especially for difficult images the resulting segmentation masks contain large zones with a high transparency. This leads to the foreground object smoothly transitioning into the background. Since this is not desired, we filter out such examples by computing a mask transparency score. The score is calculated as the percentage of non-zero pixels that have an opacity value below a certain threshold. This threshold is chosen to be an opacity of $95\,\%$, and we keep images with a mask transparency score smaller than 0.1. These pre-processing steps are applied equally for all images we collected, i.e. objects of interest and distractor objects. In the case of the object of interest, this reduces our set of images from over $21,000$ to $2,859$.

[2] See github.com/danielgatis/rembg [Last accessed on Sept. 20, 2024].

In order to analyze the impact of the image selection on the quality of the resulting dataset, we create three different datasets from the $2,859$ parcel image candidates.

Parcel2D Plain. This is the base dataset, where in addition to the above-mentioned pre-processing a selection based on mask convexity is introduced. We compute a mask convexity score, and discard all images with a mask convexity score smaller than $95\,\%$. We compute the convexity score as the quotient of the area of the biggest contour divided by its convex hull. It has a total size of $1,321$ instances of parcels.

Parcel2D CNN. We only use the annotations for the category "box" of the OpenImages dataset [Kuz+20][3], in order to train a Mask R-CNN [He+17]. We employ this Mask R-CNN to detect whether there is exactly one parcel in a scraped image. The detection score threshold is set to $95\,\%$ and we discard any images with more or less than one detected parcel. In addition, we use the same mask convexity as described for Parcel2D Plain. This dataset consists of $1,066$ instances.

Parcel2D Manual. We revised the $2,859$ candidate images manually, to only select the ones we find suitable, i.e. photos of a single parcel with a homogeneous background. The final dataset contains 854 instances.

4.2.3 Image Generation

For the generation of the final datasets, we always use the full set of distractor objects and the respective set of parcel objects. To ensure fair comparability we used the same configuration for all datasets: We sample between 1-4 objects of interests and 2-4 distractor objects. We paste these objects at a random position

[3] We use the updated dataset OpenImages V6 from https://storage.googleapis.com/openimages/web/index.html [Last accessed on Sept. 20, 2024].

onto the background while maintaining the original size of the background and introducing 2D rotations and scaling of the objects. In order to guarantee a suitable size of the objects, we limit their scale such that their (relative to the background image) longer side occupies between $15\,\%$ and $40\,\%$ of the image. In addition, we allow a maximum upscaling by $20\,\%$, since otherwise the objects potentially become overly blurry. We set a maximum Intersection over Union (IoU) of 0.5 between objects, and reattempt randomly pasting the objects onto the background, if this threshold is crossed. When a suitable arrangement of objects and distractors is found, we generate four different versions of the same image. This means, we leave the background, the objects and their positions the same, and only adjust the composition method. The following blending methods are used: no blending, gaussian blending, motion blur and Poisson blending [PGB03]. Compared to Dwibedi et al. [DMH17], we add motion blur. We generate $2,000$ training image configurations and 500 for each, validation and testing. Note that the number of images is four times that much, since we generate one image per blending method.

4.2.4 Evaluation Dataset: Parcel2D Real

In order to evaluate the usability of our approach in real world applications, we collected a dataset of parcel photos in various environments. Our validation dataset comprises 96 and the test dataset 297 images. We describe the data acquisition and the automated annotation process in the following. Note, that while our focus is on a dataset for instance segmentation, we decided to use an automated approach for the dataset generation, which inherently yields 3D annotations as well.

Data acquisition

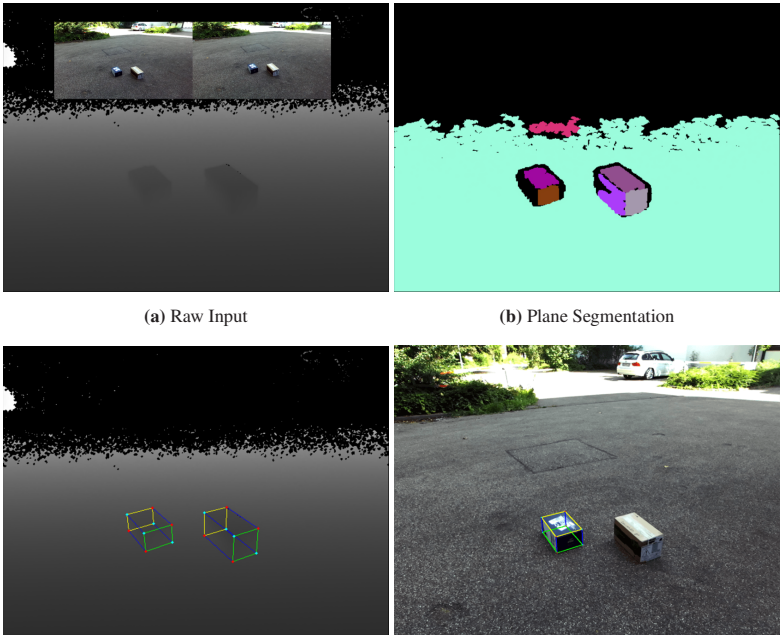
We built a custom camera rig on which we mounted a Basler Blaze time-of-flight camera and a Stereolab Zed2 stereo camera. The sensor of the Blaze and the center of the Zed2 are aligned vertically. To allow the transferral of annotations

from the depth image of the Blaze to the color images of the Zed2, we calibrated the Blaze with each, the left and the right camera of the Zed2. For the acquisition of the photos, we mounted the camera rig onto a tripod. For each image, we additionally collected the background ID and the IDs of the parcels present in the image. See Fig. 4.2 for exemplary images.

Figure 4.2: Exemplary images of the Parcel2D Real dataset. [©2022 IEEE]

Annotation generation

Starting from the captured RGB-D image as seen in Fig. 4.3a, we first applied a plane segmentation approach [Fur+18; Gri+19] (see Fig. 4.3b). To identify the ground plane, we assumed that it is close to the camera and, in comparison with other planes, relatively large. Using the ground plane, we searched all candidates for parcel top planes by computing the angle between the corresponding normal vectors. To reliably identify parcels, we discard top plane candidates based on the ground truth parcel dimensions. By projecting the remaining parcel top planes onto the ground plane, and fitting the best 3D bounding box with ground truth dimensions around the points using a RANSAC approach [FB87], we identify the final parcel annotation as exemplary shown in Fig. 4.3c. These annotations can then be projected onto the color images using the calibration information and we obtain the annotated RGB images as in Fig. 4.3d. We manually revised the dataset to not include erroneous detections.



(a) Raw Input **(b)** Plane Segmentation

(c) Annotated RGB-D image **(d)** Annotated RGB image

Figure 4.3: Visualization of the annotation generation process: (a) Raw input data from the depth and stereo camera, (b) plane segmentation result for the RGB-D image, (c) resulting annotation on the depth image, and (d) annotation that has been transferred onto the RGB image (for one image and one parcel only). [©2022 IEEE]

4.3 Evaluation" [Nau+22]

[4][We present our model and training configuration in Section 4.3.1. Next, we evaluate the importance of the image selection strategies in Section 4.3.2 and finally present an ablation that analyzes the effect of using blending methods in Section 4.3.3.]

[4] The following paragraph has been added to improve the reading flow.

4.3.1 Model Configuration

For all our experiments, we employ a ResNet-50-FPN [Lin+17] that was pre-trained on the Microsoft COCO dataset [Lin+14] as backbone. The same augmentation techniques are used during training for all datasets. For all our experiments we use Stochastic Gradient Descent with Momentum (SGD+M) with a batch size of 16 and a frozen backbone. The learning rate schedule is a cosine learning rate schedule [LH17] with an initial learning rate of 0.01 and a final learning rate of 0 after $15\,000$ iterations. Additionally, we apply a linear warm up during the first 1000 iterations. [5][We select the final model after $15\,000$ iterations and not the one with the highest Mask AP on the validation set as done in [Nau+22].]

4.3.2 Comparison of Image Selection Strategies

We analyze the influence of the three presented image selection methods, by training a ResNet-50-FPN on each of the created datasets, and subsequently evaluating their performance on Parcel2D Real that was presented in Section 4.2.4. Furthermore, we add a baseline to cover the special case when a domain-specific dataset is available. For our baseline numbers, we train on real photographs of boxes taken from the OpenImages dataset category "box". This training dataset contains 2086 instances. Note, that the OpenImages definition of box is broader than the one used for our manual image selection. The results are summarized in Table 4.1.

We see that all three methods for image selection, i.e. no image selection (Plain), image selection by Mask R-CNN and manual image selection can be used to generate suitable datasets. All resulting datasets allow a transfer from synthetic to real data as indicated by the Box AP_{75}, which is above 80 in all cases. For the case of object detection, however, training on the relevant subset of the OpenImages dataset yields the best results as implicated by the Box AP_{75}. While for the

[5] Note that this sentence has been altered from [Nau+22], to account for a slight difference in model selection.

	Box			Mask		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Parcel2D Plain	**68.5** (0.8)	**96.0** (0.6)	**85.6** (2.2)	**82.4** (0.4)	**96.4** (0.5)	**93.6** (0.4)
Parcel2D CNN	66.3 (0.3)	94.9 (0.4)	84.6 (1.0)	81.1 (0.4)	94.9 (0.4)	92.9 (0.8)
Parcel2D Manual	65.7 (0.5)	92.9 (0.4)	84.3 (1.3)	78.4 (0.3)	92.7 (0.0)	90.2 (0.4)
OpenImages	**72.9** (0.5)	**96.7** (0.4)	**94.2** (0.6)	**83.2** (0.3)	**96.7** (0.4)	**95.2** (0.5)

Table 4.1: [6][Quantitative evaluation results for bounding box detection and instance segmentation on Parcel2D Real. We repeated all training runs five times and report mean values with the standard deviation in parentheses.]

Box AP_{50}, results are comparable across the different datasets, OpenImages clearly outperforms the other datasets on Box AP_{75}. This might be due to the broader and thus, more diverse definition of the category of interest, box. The same argument can be applied to the comparison of the three image selection methods: the Plain variant performs best for object detection and segmentation and at the same time has the broadest definition of the category box, since no object-specific filtering is applied. The fact that the image domain of the training data should contain the one of the test data to get highest performance during transfer learning, was analyzed by Mensink et al. [Men+21] and can be confirmed for our application.The results for the task of image segmentation are different. All datasets have a Mask AP_{75} above 90 and thus, perform very well on the test dataset. Differences between training on the Plain dataset, the CNN dataset and OpenImages are rather small, only the manually selected dataset performs worse. The best dataset according to the Mask AP is the Plain dataset.

We cannot generalize these findings to arbitrary tasks, however, it is noteworthy that contrary to human intuition, a cautious cherry-picking of instance examples

[6] This table, including caption, has been updated from the original publication [Nau+22] by repeating all training runs five times to increase statistical meaningfulness. The performance is generally slightly lower compared to [Nau+22], but the additional analysis confirms the previously observed trends.

does not always yield the best performance. Finally, we trained the ResNet-50-FPN on both OpenImages and Parcel2D Plain combined. The results on the real test dataset are a Box AP of 72.6 and a Mask AP of 86.5. Thus, training on the combination of the two datasets is beneficial considering Mask AP.

4.3.3 Ablation Study

Since Dwibedi et al. [DMH17] and Ghiasi et al. [Ghi+21] do not agree on the importance of blending methods, we performed an ablation study to check which finding holds true in our use-case. Ghiasi et al. question the importance of using blending methods, whereas Dwibedi et al. claim, that blending methods are important for the quality of the dataset. We obtain a Box AP of 51.2 and a Mask AP of 70.9, when training on Parcel2D Manual, without using any blending methods. Since this is a considerable drop in performance, compared to the case with blending methods, we argue that blending methods are an important factor. Further, the effect that Ghiasi et al. observed, probably stems from the fact that they augment existing annotated images. These images inherently contain annotated objects where no local pasting artifacts are present in addition to pasted objects with local artifacts and thus, the model cannot focus on pasting artifacts as main visual cues.

5 Tampering Assessment for Parcels

Automated tampering assessment is crucial due to the steadily rising amount of valuable goods in supply chains. It guarantees the integrity of a parcel along the supply chain and thus, can directly affect customer satisfaction. Automating tampering detection requires two main steps: (1) we need to reliably (re-)identify parcels along their way in the supply chain. This task is demanding since its unique identifier such as a shipping label might not always be visible and distinguishing parcels based on visual cues frequently is difficult due to their homogeneous texture. (2) Appearance changes on the packaging that might stem from tampering need to be detected. Especially differences in perspective and lighting conditions of the images considerably increase the complexity associated with the task.

Since the re-identification of parcels has been studied by Clausen et al. [Cla+19], we assume in the following that the parcel has already been identified and focus on step (2), i.e. appearance change detection to identify signs of potential tampering. In accordance with our use-case last-mile delivery, our approach only takes a single RGB image as input which should be compared against a reference from a database (cf. Fig. 5.1 (a)). By estimating the parcel corner points in the image and applying a perspective transformation τ as visualized in Fig. 5.1 (b), the task can be reduced to parcel side surface matching and change detection per side surface pair (cf. Fig. 5.1 (c)). This alleviates the challenges arising from differences in viewing angles across images, however, taking the variance in lighting into account remains demanding. We evaluate different change detection approaches on our newly collected tampering detection dataset TAMPAR and reach 81% accuracy and an F1-Score of 0.83. Additional sensitivity analyses are presented to analyze

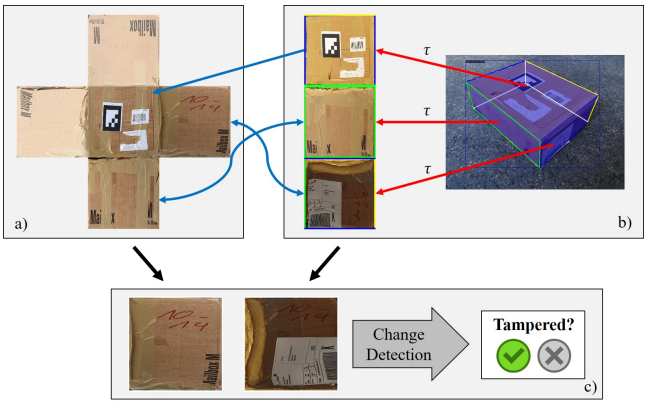


Figure 5.1: Overview of the tampering detection pipeline: We assume that the full parcel texture from a database (a) is given as a reference. We employ parcel corner point detection to the input image to generate viewpoint-invariant parcel side surface representations by applying a perspective transformation τ (b). Finally, to identify tampering we perform appearance change detection for all matching parcel side surfaces (c). [Graphic from [Nau+24], ©2024 IEEE]

the effects of tampering types, distortion and viewing angles. To summarize, the main contributions of this chapter are:

- we present an effective keypoint definition for parcels which is evaluated for parcel corner point detection,
- we introduce the novel dataset TAMPAR for TAMpering detection of PARcels which comprises more than 900 real-world images of parcels with bounding box, segmentation mask, keypoint and tampering type annotations,
- we propose and evaluate a tampering detection pipeline that combines keypoint and change detection, and
- we make our dataset and code publicly available at https://a-nau.github.io/tampar.

We review related work in Section 5.1. Subsequently, we outline our approach for parcel keypoint detection and for change detection in Section 5.2. Moreover, TAMPAR, our novel dataset for tampering detection of parcels is introduced. Section 5.3 presents the evaluation for our parcel keypoint detection approach, as well as for the full tampering detection pipeline.

Sections 5.1 to 5.3 have been previously published and are direct quotes from Naumann et al. [Nau+24], including tables and figures. These sections are marked with ❞ [Nau+24] in the respective headline.

5.1 Related Work ❞ [Nau+24]

We review related literature in logistics applications, 3D bounding box detection, keypoint estimation and change detection in the following.

Applications in Logistics. Karaca et al. [KA05] present an early approach using a stereo camera and feature matching techniques to track parcels along a conveyor belt. Clausen et al. [Cla+19] present an approach for tracking parcels inside a logistics facility. A Mask R-CNN-based [He+17] Siamese network [Bro+93] complemented with their so-called feature improver head is used to re-identify parcels. They create a manually labeled dataset of 3,306 images taken by 37 different cameras with a total of 14,248 parcels. The evaluation shows that currently around $81\,\%$ of parcels are tracked correctly. For more details on literature regarding re-identification we refer to Ye et al. [Ye+22] and Khan et al. [KU19]. Naumann et al. [Nau+20] work towards parcel side surface segmentation. By combining plane segmentation [Liu+19] and contour detection [Can86; SRS20], they present an approach to refine parcel side surface segmentation masks without relying on any task-specific training data. Naumann et al. [Nau+23a] tackle the problem of estimating the 3D shape of potentially damaged parcels from a single RGB input. They extend Cube R-CNN [Bra+23] by an iterative mesh refinement [GMJ19] and present Parcel3D, a dataset comprising over 13,000 images

of cuboid-shaped and damaged parcels with full 3D annotations. Noceti et al. [NZO18] present a multi-camera system for damage and tampering detection in postal supply chains. Damages are detected by finding the parallelepiped which best aligns with the captured images. For tampering detection a Histogram of Oriented Gradients (HOG) [DT05] for the parcel side surfaces is used. Rotation invariance is accomplished by considering all possible rotations with histogram intersection as similarity measure. Tampering is reported when the similarity of two feature vectors is below a certain threshold. Other works focusing on parcels consider synthetic training data generation [Nau+22], tracking inside a moving truck [Hu+21a] and depalletization [Arp+20; Chi+20]. Finally, Naumann et al. [Nau+23b] present a detailed overview of computer vision applications in transportation logistics and warehousing.

3D Bounding Box Detection. Dwibedi et al. [Dwi+16] present an early deep learning-based approach to estimate the 3D bounding box of cuboid-shaped objects. Generally, 3D bounding box detection is a common task for autonomous driving [Arn+19]. Approaches often rely on only estimating yaw, since they can exploit the fact that vehicles are driving on the road. Li et al. [Li+20] exploit 2D/3D correspondences by estimating keypoints of cars to improve 3D bounding box detection. Rui et al. [Rui+20] introduce a framework for vehicle recognition from a single RGB image. They estimate a 3D bounding box which is used to compute normalized views for the front, side and roof view of a car by applying a perspective transformation. This information is fused with region-aligned features of the respective region of interest to estimate the vehicle model.

Keypoint Detection. Lots of research tackling keypoint estimation considers monocular human pose estimation, which is reviewed by Chen et al. [CTH20] and Chen et al. [Che+22]. Dörr et al. [Dör+21] treat the problem of packaging structure recognition. The goal is to identify the number, type and arrangement of small load carriers on a uniformly packed transport unit from a single RGB image. They extend CornerNet [LD18] to leverage keypoint estimation to detect objects based on four arbitrary corner points.

Change Detection. To detect signs of tampering, after re-identification, change detection is necessary. Change detection is most commonly applied for remote-sensing and street views and reviewed by Shi et al. [Shi+20]. A dataset for change detection in industrial environments has been presented by Park et al. [Par+21]. Furthermore, Park et al. [Par+22] propose the novel change detection approach SimSaC which is targeted towards industrial use-cases. SimSaC relies on dual task learning and exploits both, dense correspondence and mis-correspondence to increase robustness when encountering imperfect matches.

While Noceti et al. [NZO18] also tackle the problem of tampering detection, they focus on a constrained environment with calibrated background, constant illumination and a multisensory setup. In contrast to that, our approach does not have any such constraints and relies just on a single RGB image as input. Consequently, ours is the only approach suitable for scenarios such as last-mile delivery and cannot be fairly compared to the work by Noceti et al. [NZO18]. Furthermore, we rely on existing keypoint and change detection approaches and strive to combine them efficiently, however, we do not aim to develop novel approaches in these areas.

5.2 Approach” [Nau+24]

We present our approach for parcel keypoint detection in Section 5.2.1 and for change detection in Section 5.2.2. Details on our novel dataset TAMPAR are given in Section 5.2.3.

5.2.1 Parcel Keypoint Detection

We use a Mask R-CNN [He+17] with keypoint head and a ResNet-50-FPN [He+16; Lin+17] backbone for our experiments. This choice is motivated by the fact, that we do not focus on improving keypoint detection techniques, but

rather want to demonstrate the usefulness of well-established baselines for the use-case of parcel corner detection.

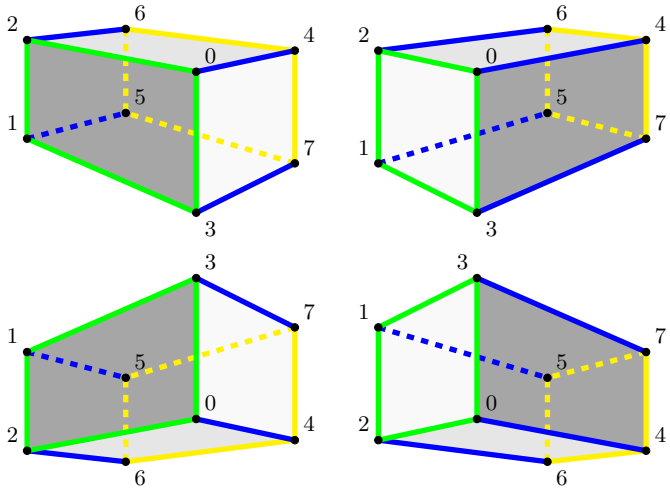


Figure 5.2: Visualization of the consistent and unambiguous keypoint ordering for a cuboid without well-defined front and back across different viewing angles. We highlight the front side in green and the back side in yellow. [©2024 IEEE]

One key challenge for this use-case is to identify an unambiguous keypoint ordering which works well with Artificial Neural Networks since there are several options for ordering keypoints of a parcel. In contrast to the common application of 3D bounding box detection for autonomous driving, where vehicles have a well-defined front and back side, there is no such notion for parcels. To have a consistent, unambiguous keypoint ordering with explicit visual cues, we proceed as follows. We assume, that three parcel side surfaces are visible in each image and define the front of a parcel by choosing the visible parcel side surface whose normal aligns best with a left- and front-facing vector, i.e. $(x, y, z) = (1, 0, -0.5)$. From this, we derive our keypoint ordering definition, which is visualized in Fig. 5.2 and described in the following. We denote the number of visible α and invisible β parcel side surfaces that intersect in keypoint k_i as k_i=(α, β). On the front side (highlighted in green in Fig. 5.2), we define the keypoints:

- k_0 = (3, 0): point of intersection of the three visible parcel side surfaces, which is located inside the convex hull of the parcel.
- k_1=(1, 2): joint point of the two invisible parcel side surfaces, where only two visible parcel edges intersect.
- k_2=(2, 1), leftmost: leftmost point of the remaining points, where three visible parcel edges intersect.
- k_3=(2, 1), rightmost: remaining point, which is the rightmost point that belongs to two visible parcel side surfaces and one invisible one.

The backside (highlighted in yellow in Fig. 5.2) of the parcel is the one across from the front side, and we define the keypoint order as follows:

- k_4=(2, 1): point that is part of two visible parcel side surfaces and thus, at this point three visible parcel edges intersect.
- k_5=(0, 3): self-occluded keypoint, which is the point of intersection of the three invisible parcel side surfaces.
- k_6=(1, 2), leftmost: leftmost point of the remaining points, where two visible parcel edges intersect.
- k_7=(1, 2), rightmost: remaining point, which is the rightmost point where two visible edges intersect.

This keypoint ordering is used for training and evaluating corner point detection in the following. Note, that it is not invariant to horizontal, but only to vertical flipping of the image. Furthermore, technically, estimating seven keypoints would be sufficient to infer all eight, however, we want to show that the estimation even works for the self-occluded keypoint k_5. The information on the seven visible keypoints can be utilized to compute viewpoint-invariant parcel side surface representations by applying a perspective transformation. This, in turn, enables the composition of parcel texture mappings as visualized in Fig. 5.1 (a).

5.2.2 Change Detection

In our use-case, we assume that a postman takes a single image of a parcel which seems suspicious of potential tampering. First, the parcel keypoints are extracted and the viewpoint-invariant parcel side surfaces of size 400×400 pixels are computed as described in Section 5.2.1 and visualized in Fig. 5.1 (b). By exploiting this information, we can reduce the task of tampering detection of parcels to comparing fronto-parrallel parcel side surface representations. If one parcel side surface has been tampered with, the parcel is considered tampered.

While the usage of viewpoint-invariant representations alleviates the problem of perspective distortion, change detection remains challenging since image alignment issues cannot fully be resolved, and additionally, the lighting might vary significantly (cf. Fig. 5.1 (c)). To cope with these issues, we use SimSaC [Par+22]. SimSaC is a recent approach for robust change detection with imperfect matches. It estimates scene flow using correspondence maps at the same time as change masks by exploiting mis-correspondences. This enables robustness against geometric transformations and differences in lighting.

We benchmark SimSaC against several baselines, each combining an image homogenization approach and a similarity metric. For image homogenization, we utilize (cf. Fig. 5.3)

- *DexiNed*: Dense EXtreme Inception Network for Edge Detection [SRS20]
- *Canny*: Adaptive Canny edge detection [Can86; JN12]
- *Laplacian*: Laplacian filter
- *Mean Channel*: Per-channel mean alignment

As image similarity metrics, we consider

- Learned Perceptual Image Patch Similarity (LPIPS) [Zha+18],
- Structural Similarity (SSIM) [Wan+04],
- Multiscale Structural Similarity (MS-SSIM) [WSB03],

- Complex Wavelet Structural Similarity (CW-SSIM) [Sam+09],
- HOG [DT05] feature similarity [1], and
- Mean Absolute Error (MAE).

(a) None **(b)** SimSaC **(c)** DexiNed **(d)** Canny **(e)** Laplac. **(f)** Mean Ch.

Figure 5.3: Examples of the different image homogenization methods before (top) and after (bottom) tampering. Note that SimSaC [Par+22] is the only approach that localizes potential tampering and directly outputs change maps. [©2024 IEEE]

A change is detected when the input and reference parcel side surface image after applying the image homogenization to both, have a low image similarity. Suitable thresholds for image similarity will be determined in Section 5.3.2.

5.2.3 Dataset

Our dataset resembles a use-case with multisensory setups within logistics facilities and a simple cell phone camera during the last-mile delivery. More precisely, we assume that multiple cameras are used to capture and segment all five visible parcel side surfaces in logistics facilities. Note that we also suppose that the side surface with the unique identifier is always visible, which means that the opposing side surface is never visible. Subsequently, the parcel ID and texture map, as visualized in Fig. 5.1 (a), are saved to a database. Finally, a single RGB

[1] We use 9 orientation bins, 8 pixels per cell and 2 cells per block.

image of a parcel with suspected tampering is taken during last-mile delivery and compared against its high-quality reference texture.

To generate a suitable dataset for this use-case, we proceed as follows. We use ArUco markers to uniquely identify parcels and the spatial relationships between their side surfaces. The parcel textures for the database are generated by taking several images of the parcel in its original state, i.e. without tampering. By manually labeling the parcel corner points, we automatically generate the full parcel texture by applying perspective transformations. Subsequently, we apply different types of tampering to three out of the five relevant parcel side surfaces. While real-world tampering attempts focus on a single parcel side surface, our dataset design enables a more diverse analysis of tampering detection by considering a larger number of tampering examples. As mentioned before, transferring side surface tampering to the parcel level is straightforward. We consider three different types of tampering flags, each with an *easy* and a *hard* to detect variant:

- *Label*: Adding a new shipping label (*easy*) or transportation hints (*hard*)
- *Tape*: Adding new tape, which covers more than 50% of the longer side (*easy*), or less than 25% of the shorter side (*hard*)
- *Writing*: Adding manually written text, using a pen with 5-$15\,\mathrm{mm}$ (*easy*) or $1.5-3\,\mathrm{mm}$ of width (*hard*)

Note that adding written text usually would not be considered tampering. However, we strive to detect diverse appearance changes to reliably flag parcels for manual inspection. In total, we collect and annotate 296 images of 10 parcels for the training/validation and 614 images of 20 parcels for the test set. Since each image contains three visible parcel side surfaces, TAMPAR comprises 888 images for training/validation and 1842 images for testing change detection. The main difference to existing datasets such as Parcel3D [Nau+23a] and Parcel2D Real [Nau+22] is that we have paired images of the same parcel across different points in time, i.e. before and after tampering.

5.3 Evaluation” [Nau+24]

We first evaluate keypoint detection for parcel corners separately in Section 5.3.1. Subsequently, we evaluate the considered change detection approaches isolated and in combination with keypoint estimation in Section 5.3.2. Furthermore, we present a sensitivity analysis on the influence of the tampering type, lens distortion and viewing angles.

5.3.1 Parcel Corner Point Estimation

For all experiments, we use a ResNet-50-FPN [He+16; Lin+17] that was pre-trained on MS COCO [Lin+14] as backbone and freeze its weights at stage four. We use Stochastic Gradient Descent with Momentum (SGD+M) with a batch size of 16 and a cosine learning rate schedule [LH17]. The initial learning rate is set to 0.001 and the final learning rate to 0 after $10\,000$ iterations. Moreover, a linear warm-up during the first 1000 iterations is applied.

Training is always performed on the synthetic dataset Parcel3D [Nau+23a] which contains cuboid-shaped and damaged parcel images. For the evaluation, we consider synthetic and real-world data in the following. We evaluate bounding box detection, instance segmentation and keypoint detection, and summarize the quantitative results in Table 5.1.

For the evaluation of keypoint detection using Keypoint AP[2], it is necessary to define κ_i for each keypoint. This value is usually obtained by comparing redundantly annotated images to infer each keypoints' annotation precision. Since no redundantly annotated images are available, we select $\kappa_5 = 0.1$ for the self-occluded and $\kappa_i = 0.05,\ i \in \{0, 1, 2, 3, 4, 6, 7\}$ for the visible keypoints, which is close to the κ for human hips (0.107) and human wrists (0.062), respectively [Lin+14]. We argue that human wrists are a suitable approximation because the

[2] See https://cocodataset.org/#keypoints-eval for details.

keypoints for Parcel3D and Parcel2D Real are computed from 3D bounding boxes, frequently leading to a misalignment between the parcel corners in the image and the actual annotated keypoints. This misalignment is also present for damaged parcels, where the keypoints correspond to the ones of the pristine version of the parcel.

	Box		Mask		Keypoint	
Dataset	AP	AP_{75}	AP	AP_{75}	AP	AP_{75}
Parcel3D	93.62 (0.1)	98.46 (0.2)	97.54 (0.2)	98.58 (0.3)	88.80 (0.2)	94.06 (0.2)
Parcel2D Real	84.88 (0.2)	97.28 (0.1)	85.02 (0.2)	96.92 (0.6)	75.76 (0.5)	85.36 (1.2)
TAMPAR (ours)	96.38 (0.2)	99.72 (0.5)	98.94 (0.2)	99.70 (0.5)	97.18 (0.5)	99.12 (0.4)

Table 5.1: Quantitative performance analysis of the ResNet-50-FPN for bounding box detection, instance segmentation and keypoint detection. We repeated all trainings five times and report *mean (standard deviation)*.

5.3.1.1 Synthetic Data

The quantitative results from Table 5.1 indicate excellent performance for bounding box detection and instance segmentation, with a Box AP of 93.62 and a Mask AP of 97.54. Likewise, keypoint detection achieves strong results with a Keypoint AP of 88.80.

Qualitative examples are presented in Fig. 5.4. For intact, i.e. cuboid-shaped, parcels keypoint detection enables computing high-quality fronto-parallel views of the parcel side surfaces as can be seen in Fig. 5.4a. Strong distortions of parcel side surface views, however, cannot be recovered and lead to low-quality representations, which are challenging to use for tampering detection. In the case of damaged parcels, the representations' quality strongly depends on the degree of deformation (cf. Fig. 5.4b). Strong deformations also impede tampering detection. Problematic cases can include imprecise or missing detections (cf. Fig. 5.4c).

5.3.1.2 Real Data

Due to the fact that training was only performed on the synthetic training dataset Parcel3D [Nau+23a], a domain gap occurs when evaluating on the two real-world datasets Parcel2D Real [Nau+22] and TAMPAR. This domain gap manifests itself in the generally lower performance on Parcel2D Real compared to the evaluation on synthetic data, as seen in Table 5.1. At the same time, performance on TAMPAR is higher, presumably due to the simpler nature of the dataset - all images are high-quality and show only a single parcel in the center. Performance for bounding box detection and instance segmentation remains high with a Box AP of $84.88/96.38$ and a Mask AP of $85.02/98.94$, on Parcel2D Real and TAMPAR, respectively. The same holds true for the performance of keypoint detection, which reaches 75.76 and 97.18 Keypoint AP.

Quantitative inspection of the prediction results confirms the suitability of Parcel3D and our proposed keypoint ordering. Especially for cuboid-shaped parcels, as visualized in Fig. 5.5a, results look very promising for applications in tampering detection. Furthermore, we evaluate our approach on images of damaged parcels without ground truth annotations (cf. Fig. 5.5b). These qualitative impressions also underpin the suitability of our approach, however, detecting keypoints accurately for damaged parcels seems to be a more difficult task. Examples of failed detections include missing and imprecise keypoint localizations, as visualized in Fig. 5.5c.

5.3.1.3 Sensitivity Analysis: Lens Distortion

We investigate the influence of barrel distortion according to

$$r_{\text{src}} = r_{\text{dist}} \cdot \left(A \cdot r_{\text{dist}}^3 + B \cdot r_{\text{dist}}^2 + C \cdot r_{\text{dist}} + D \right)$$

with r_{src} being the radial distance from the image center in the input image, and r_{dist} the one, in the distorted output. We analyze six different settings, which are visualized in Fig. 5.6 by creating distorted dataset versions with parameter

$A \in [-0.08, -0.04, -0.02, 0.04, 0.08, 0.16]$, $B = 0$, $C = 0$, and $D = 1.0$. Note that these datasets can be smaller in size, since we discard instances if the distortion corrupted the annotations (e.g. keypoints lie outside the image) or the ArUco detection. Results in Fig. 5.7 indicate that instance segmentation performance is robust w.r.t. distortion effects. While keypoint detection only degrades for pincushion distortions ($A < 0$), bounding box detection results are also affected by strong barrel distortions ($A > 0$).

5.3.2 Tampering Detection

We evaluate change detection in isolation by using the ground truth keypoint annotations, as well as in a combined system on our novel dataset TAMPAR by using keypoint predictions from Section 5.3.1. Furthermore, we analyze the influences of tampering types, lens distortion, and viewing angles.

5.3.2.1 Pipeline Evaluation

Considering the input and reference parcel we first perform marker-based side surface matching. Subsequently, we apply the image homogenization to both (input and reference side surface view) and compute their image similarity using all metrics mentioned in Section 5.2.2. We denote the combination of image homogenization *Method A* and similarity *Metric B* as *(Method A, Metric B)*, and seek to determine the best image similarity metrics and corresponding thresholds. SimSaC [Par+22] poses a special case since it uses the input and reference image to output change maps. This enables localization of tampering, which is advantageous in practice, however, not evaluated in this work. Instead, we compare the binary change map to a black image (i.e. the change map corresponding to no changes) to compute image similarity. We summarize the evaluation results using simple thresholding by training a decision tree of depth one per method using all similarity metrics as input in Table 5.2.

Method	Metric	Accuracy	Precision	Recall	F1-Score	ROC-AUC
None	LPIPS/MS-SSIM	0.66/0.65	0.66/0.65	0.91/0.93	0.76/0.76	0.60/0.58
SimSaC	LPIPS/MAE	**0.81**/**0.80**	**0.91**/**0.93**	0.76/0.72	**0.83**/**0.81**	**0.82**/**0.82**
DexiNed	HOG/SSIM	0.60/0.62	0.60/0.63	**1.00**/0.91	0.75/0.74	0.48/0.54
Canny	MS-SSIM/SSIM	0.60/0.60	0.62/0.61	0.91/0.91	0.74/0.73	0.52/0.52
Laplacian	LPIPS/LPIPS	0.65/0.68	0.71/0.71	0.72/0.80	0.71/0.75	0.64/0.65
Mean Ch.	LPIPS/MS-SSIM	0.63/0.65	0.62/0.65	0.99/**0.94**	0.76/0.77	0.53/0.58

Table 5.2: Quantitative performance analysis of the tampering detection using a decision tree with depth one. The metric indicates the selection for thresholding during the training of the decision tree. We report metric names and scores for *predicted / ground truth* keypoints.

Results in Table 5.2 using predicted keypoints show that *(SimSaC, LPIPS)* yields the best performance and reaches 0.81 accuracy and an F1-Score of 0.83. The by far highest precision is also achieved by *(SimSaC, LPIPS)*, which indicates cautious change detection for our use-case. The highest recall is reached by *(DexiNed, HOG)* and *(Mean Ch., LPIPS)*, however, at the cost of precision. Performance differences between using predicted and ground truth keypoint positions are comparatively small due to the high accuracy of the keypoint detection (cf. Table 5.1).

5.3.2.2 Sensitivity Analysis: Tampering Types

The analysis of performance differences across tampering types in Table 5.3, shows that *labels* can be detected most reliably, while *tape* and especially *writing (hard)* are more difficult to recognize. Surprisingly, when detecting *writing* performance deteriorates when using ground truth keypoint annotations. One potential reason for this might be, that inaccurate keypoints enlarge the region of interest unproportionally.

Tampering	Label		Tape		Writing	
Type	easy	hard	easy	hard	easy	hard
Number of Samples	606	570	462	546	624	498
Recall (Pred. Keypoints)	1.00	1.00	0.58	0.48	0.87	0.52
Recall (GT Keypoints)	1.00	0.99	0.59	0.49	0.80	0.36

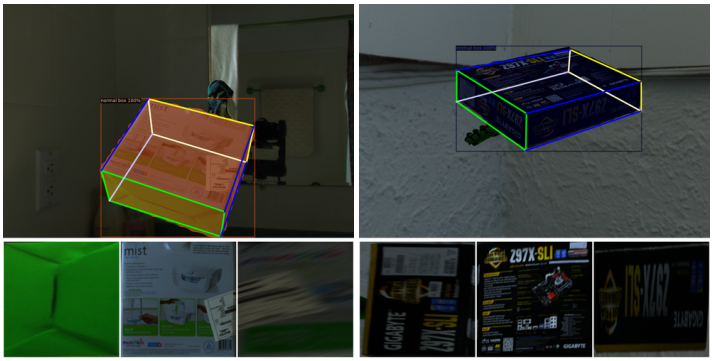
Table 5.3: Sensitivity analysis on the performance differences across tampering types using *(SimSaC, LPIPS)*.

5.3.2.3 Sensitivity Analysis: Lens Distortion

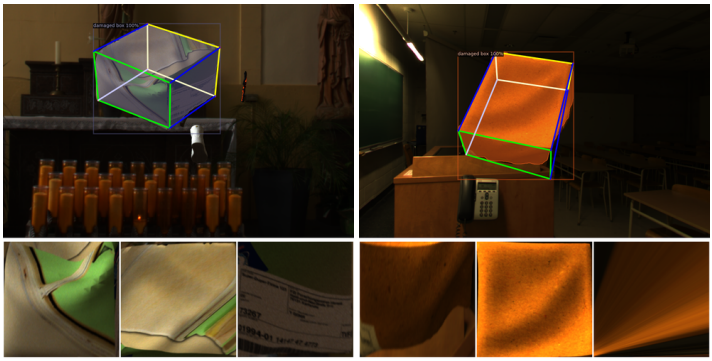
We analyze the influence of six different degrees of distortion (cf. Fig. 5.6) on the tampering detection quality using predicted keypoints and *(SimSaC, LPIPS)*. These distortions imply that our simple perspective transformation cannot accurately create normalized side surface views and the change detection approach needs to handle these inaccuracies. The results in Fig. 5.8 suggest robustness w.r.t. distortions, with a slight negative trend for distortions with distortion strength $A > 0$. This is in line with the fact, that our approach can cope with lens distortion effects across the two real-world dataset TAMPAR and Parcel2D Real, while being trained on different, synthetic data.

5.3.2.4 Sensitivity Analysis: Viewing Angles

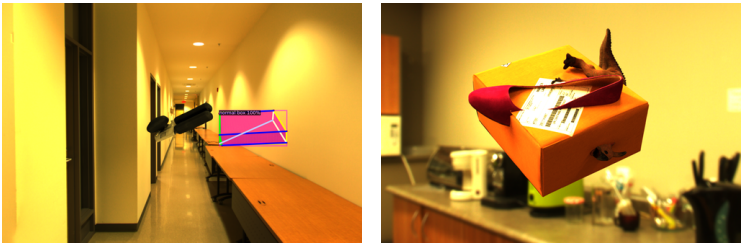
We approximate the viewing angle per parcel side surface by considering the angle between the x- and y-axis, and the polygon spanned by the four side surface corner points. No clear trend emerges from this analysis in Fig. 5.9, which suggests that our approach is robust w.r.t. a reasonable spectrum of viewing angles. Note, however, that TAMPAR does not feature extreme viewing angles. Due to the strong distortions under such viewing angles, we expect the performance of tampering detection to degrade heavily.



(a) Intact Parcels

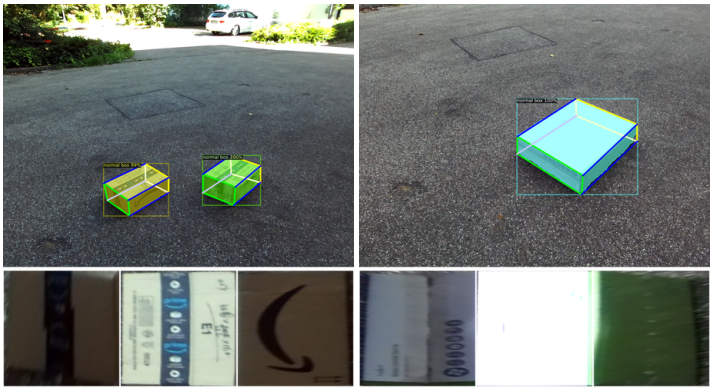


(b) Damaged Parcels

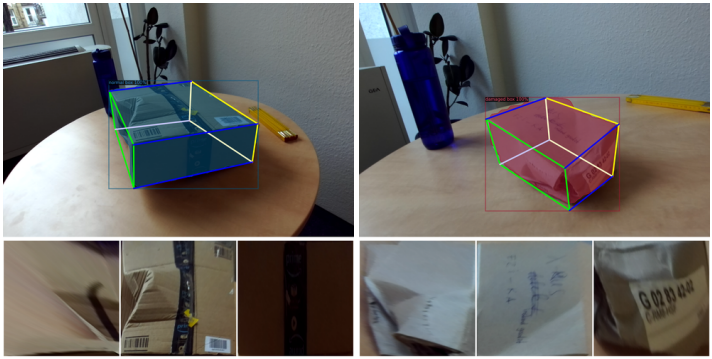


(c) Problematic Cases

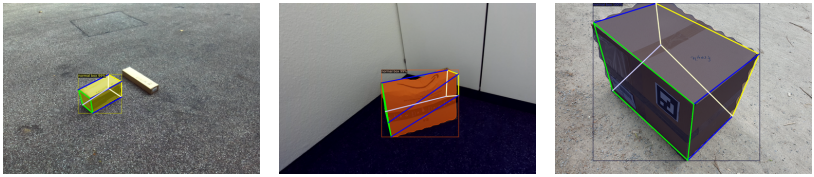
Figure 5.4: Exemplary qualitative results for synthetic parcels. [©2024 IEEE]



(a) Intact Parcels



(b) Damaged Parcels



(c) Problematic Cases

Figure 5.5: Exemplary qualitative results for real parcels. [©2024 IEEE]

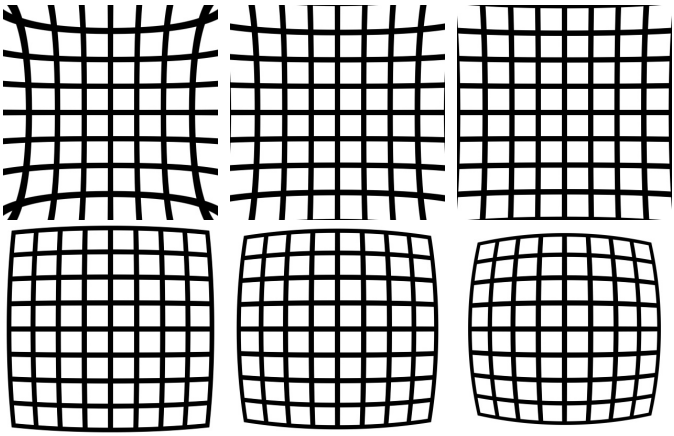


Figure 5.6: Visualization of the investigated distortion effects with parameter $A \in [-0.08, -0.04, -0.02, 0.04, 0.08, 0.16]$, $B = 0$, $C = 0$, and $D = 1.0$. [©2024 IEEE]

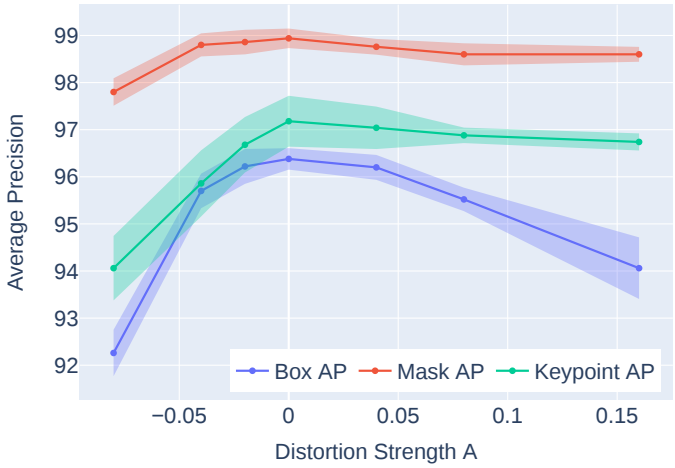


Figure 5.7: Quantitative performance analysis of the ResNet-50-FPN under different types of lens distortion. We repeated all trainings five times and report mean values with standard deviations as error boundaries. [©2024 IEEE]

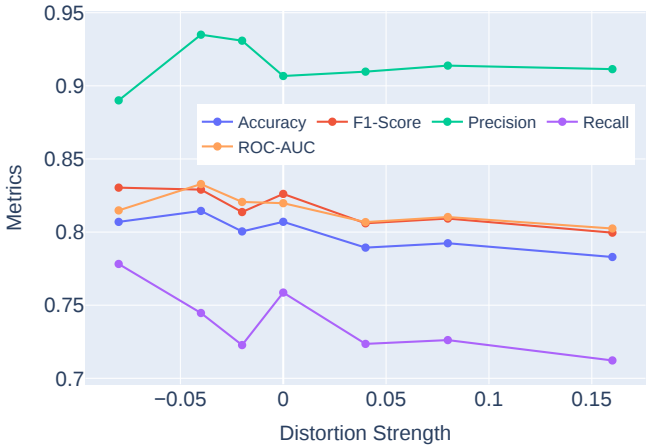


Figure 5.8: Sensitivity analysis for tampering detection w.r.t. to the distortion strength A using pred. keypoints and *(SimSaC, LPIPS)*. [©2024 IEEE]

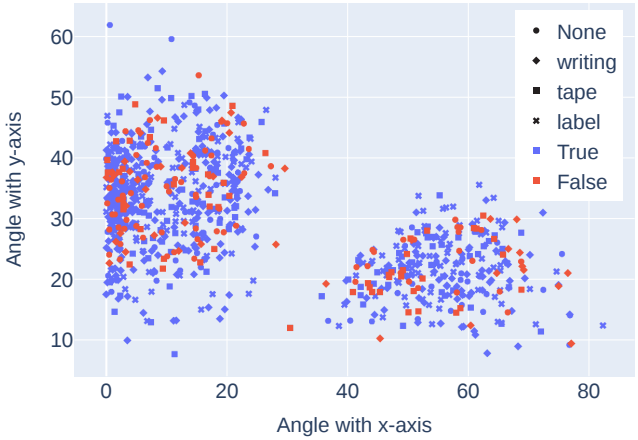


Figure 5.9: Sensitivity analysis for tampering detection w.r.t. to the viewing angle per side surface using predicted keypoints and *(SimSaC, LPIPS)*. Tampering types are encoded with different geometries and the prediction correctness with color-coding. [©2024 IEEE]

6 Damage Assessment for Parcels

In this chapter, we tackle the task of damage assessment for parcels. We limit ourselves to 3D shape reconstruction of parcels, and, thus, consider only shape deformations. Other types of common damages, such as water damage, are not treated in this work. Note that it is also not possible to reliably infer information about the state of the transported item just from a detailed inspection of the packaging.

We present our model CubeRefine R-CNN for damage assessment, which extends the 3D bounding box detection approach Cube R-CNN [Bra+23] by an iterative mesh refinement [Wan+18; GMJ19] (cf. Fig. 6.1). Since this approach only relies on a single RGB image as input, it is suitable for dynamic environments such as for postmen or clients during last-mile delivery. Moreover, as our approach simultaneously estimates the current, potentially deformed shape and the original shape of the pristine parcel (in the form of the 3D bounding box), it enables a detailed 3D analysis of parcel deformations. Furthermore, the estimated 3D bounding box could be leveraged to generate viewpoint-invariant parcel side surface views for tampering detection as suggested by [Nau+24]. Due to the lack of suitable datasets, we introduce Parcel3D: a novel synthetic dataset with over 13 000 images of intact (cuboid-shaped) and damaged parcels. The main contributions of this chapter are

- we propose CubeRefine R-CNN, a novel architecture that combines 3D bounding box detection with an iterative mesh refinement for single image 3D shape reconstruction,

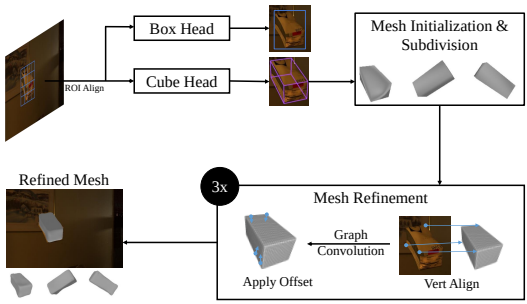


Figure 6.1: Overview of the CubeRefine R-CNN architecture: Using Cube R-CNN's *Cube Head* [Bra+23] we first estimate a 3D bounding box from the RGB input image. By converting the 3D bounding box into a mesh and applying iterative subdivision, we obtain an initial mesh reconstruction. This initial reconstruction is iteratively refined in three stages as proposed by [Wan+18]. [Graphic from [Nau+23a], ©2023 IEEE]

- we present a novel dataset, called Parcel3D, comprising synthetic images of intact and damaged parcels with full 2D and 3D annotations, and
- we make our dataset and code publicly available at https://a-nau.github.io/parcel3d.

The remainder of this chapter is structured as follows. We present an overview of the related literature in Section 6.1. Subsequently, we present details on the dataset generation approach in Section 6.2. Section 6.3 outlines, the proposed Artificial Neural Network (ANN) architecture CubeRefine R-CNN and finally, Section 6.4 presents the evaluation on synthetic and real data.

Sections 6.1 to 6.4 have been previously published and are direct quotes from Naumann et al. [Nau+23a], including tables and figures. These sections are marked with ❞ [Nau+23a] in the respective headline.

6.1 Related Work" [Nau+23a]

To the best of our knowledge there is no prior work on shape reconstruction from single images in transportation logistics and warehousing. We review literature on applications in logistics, cuboid reconstruction from RGB images and finally, 3D reconstruction of arbitrary objects from single images in the following.

Applications in Logistics. There is work on 2D segmentation of parcels [Nau+20; Nau+22], packaging units [MGF20; May+20] and packaging structure recognition [Dör+20b; Dör+21]. Moreover, there has been research on 3D reconstruction from RGBD images [Li+12; Pra+15; Son+17a; Arp+20] and from multiple views [NZO18]. 3D reconstruction by using RFID technology has been explored in [Bu+17]. Damage and tampering detection has been tackled by Noceti et al. [NZO18] in a constrained multi-camera setup. Tampering is detected by comparing normalized parcel side surfaces and damage detection by fitting a parallelepiped across multiple views. For an in-depth review on computer vision applications in logistics, we refer to Naumann et al. [Nau+23b].

Cuboid reconstruction. Cuboid reconstruction from single RGB images by identifying its 8 corner points in 2D has been tackled in the literature. Approaches are class agnostic, meaning that diverse object categories are considered as either cuboid or not. Xiao et al. [XRT12] present such an approach in the pre-deep learning era that leverages corner and edge detection techniques. After the rise of deep learning, also cuboid reconstruction was tackled with ANNs. Dwibedi et al. [Dwi+16] present an approach to estimate the position of the 8 cuboid keypoints using deep learning. A similar line of work is concerned with 3D bounding box estimation for cars [FZL19; LYL21; Kum+22], which is reviewed in-depth by Ma et al. [Ma+23]. Note, that by assuming that cars are driving on the road, rotation estimation can be reduced to yaw estimation. Approaches leverage geometric priors by requiring consistent vanishing points [Rui+20] and by imposing 2D/3D

consistency [Li+20]. Recently, Brazil et al. [Bra+23] introduced a large benchmark for 3D object detection, which combines several existing datasets. Moreover, they present a simple and effective model for 3D object detection, called Cube R-CNN.

Single RGB image 3D reconstruction. There are many approaches for general image-based 3D reconstruction without a confinement to an object type. While the input for many approaches is a single RGB image, the output varies: representations based on voxels [Cho+16; Xie+19; Yan+21], meshes [Kan+18; Wen+19; GMJ19] and pointclouds [FSG17; GWM18] are common. In addition to that, implicit representations [Mes+19; Zak+21] have been introduced. Most reconstruction approaches focus on single instances, either by considering only images with a single instance or by employing 2D segmentation. More recently, also NeRFs [Mil+20] have been used to tackle single-view reconstruction [Mul+22]. Apart from supervised approaches, there has been work on 3D reconstruction from 2D supervision [Kan+18], unpaired image collections [DP22] and unsupervised reconstruction [ID18; Nav+20; WRV20; Hu+21b], since training data with ground truth 3D annotations is difficult and costly to obtain. Han et al. [HLB21] present an overview of approaches from the deep learning era that leverage either single or multiple RGB images for 3D reconstruction. The reviews of Fu et al. [Fu+21] and Khan et al. [Kha+22] focus explicitly on single image 3D reconstruction.

We introduce the new dataset Parcel3D to enable research on image-based 3D reconstruction in the domain of logistics. Furthermore, we leverage the existing general 3D object detection architecture Cube R-CNN [Bra+23] and extend it by an iterative mesh refinement. Adding the iterative mesh refinement is necessary, since 3D object detection approaches are not suitable for damage detection and analysis. In contrast to other 3D reconstruction approaches, CubeRefine R-CNN directly enables comparing the original shape of a cuboid-shaped object with its current state, which is crucial for damage quantification.

6.2 Dataset" [Nau+23a]

We present details on the generation of our synthetic dataset Parcel3D and start by describing the automatic selection process for suitable Google Scanned Objects (GSO) [Dow+22] object models in Section 6.2.1. Next, the approaches to generate data for damaged parcels and for new textures are presented in Section 6.2.2 and Section 6.2.3, respectively. Finally, we present details on the rendering in Section 6.2.4.

6.2.1 Model Selection

We use GSO as a base dataset, since it has a wide variety of realistic 3D models. We create a new subset of the GSO dataset that is tailored towards our use-case in transportation logistics and warehousing by automatically selecting relevant models based on their shape. This filtering is done by evaluating each model's similarity with a surrounding cuboid. We initialize a template mesh from the surrounding cuboid and use the Chamfer Distance t_C and Normal Consistency t_N between this template mesh and the model mesh for comparison.
We divide the models in three categories using empirically determined thresholds for both similarity metrics. Models with $t_C \leq 0.1$ and $t_N \geq 0.9$ are chosen as cuboid models due to their high resemblance with the desired shape. We refer to these picked models by $\mathbb{G}_\mathrm{P}$. The second threshold of $t_C \leq 0.5$ and $t_N \geq 0.8$ identifies objects that are not closely related to a cuboid in shape, yet similar. These models are denoted $\mathbb{G}_\mathrm{R}$. All other models are referred to by $\mathbb{G}_\mathrm{D}$. We use models from $\mathbb{G}_\mathrm{D}$ as distractor objects, which we also render into images to prevent overfitting on rendering artifacts [DMH17]. The models from $\mathbb{G}_\mathrm{R}$ are not used as distractors, since their resemblance in shape with a cuboid might be confusing. The subset $\mathbb{G}_\mathrm{D}$ contains 750 models, $\mathbb{G}_\mathrm{R}$ contains 71 models and $\mathbb{G}_\mathrm{P}$ contains 209 models. Exemplary instances for each of the three categories are visualized in Fig. 6.2.

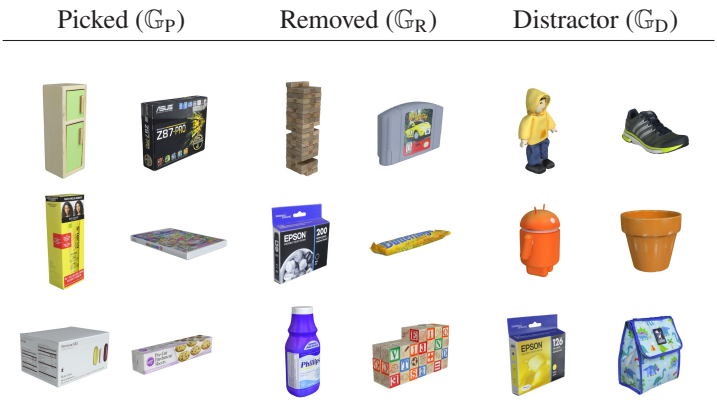


Figure 6.2: Samples of the three object model subsets of the GSO dataset [Dow+22] that were generated based on the models' similarity with a cuboid. [©2023 IEEE]

Since there are very similar models within $\mathbb{G}_P$, we combine the models into 66 groups. The grouping is done automatically by using brand and category names, since the GSO dataset contains similar object models as seen in Fig. 6.3 for the example of Pepsi cartons.

Figure 6.3: Visualization of the similarity between certain models. [©2023 IEEE]

6.2.2 Model Generation

Since we obtain only 209 suitable models from the GSO dataset, we generate 10 scaled versions for each of them. The scaling is done for each of the three dimensions separately by sampling a scaling factor from a triangular distribution

with lower limit 0.5, upper limit 2 and mode 1. These models make up the subset of intact boxes.

This method for dataset generation is suitable for intact parcel recognition, however, automatically identifying suitable models for damaged boxes within the GSO or other datasets such as ShapeNet [Cha+15], is difficult. Thus, we automatically generate models for damaged boxes using physics-based simulation in Blender. For each simulation, we start by randomly sampling a base model from the previously generated subset of intact boxes. The chosen model is then simulated to be falling onto a rigid ground as seen in Fig. 6.4. Soft body simulation is used to allow deformations during the collision. We sample falling height, angle and soft body physics parameters randomly within empirically determined ranges to obtain a wide variety of deformations. Only models from timesteps that have between 75% and 90% of their original volume are chosen as suitable models for damaged parcels. These thresholds ensure that models have at least a certain degree of deformation, while not allowing extreme changes in appearance. Furthermore, we use a RANSAC algorithm [FB87] to find the best rigid transformation between the original, cuboid-shaped model and the deformed model during simulation, to track the position and rotation of the object. Note, that this is necessary, since Blender does not incorporate the tracking of objects during a soft body simulation. Using this information we are able to identify the area of impact with the strongest deformation, which allows us to render damaged parcels such that the impacted area is visible. Finally, we apply a smoothing filter in Blender to the selected models.

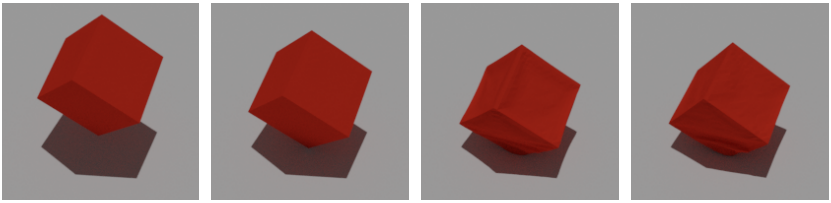


Figure 6.4: Visualization of the collision for damaged parcels using soft body simulation in Blender. [©2023 IEEE]

6.2.3 Texture Generation

In order to obtain more variance in the textures of the models and to bias the training data towards cardboard, we generate new textures. We use a cardboard shader in Blender[1], to generate a dataset of 230 cardboard textures. These textures replace the original texture of the model with a probability of 0.6, and an example is shown in Fig. 6.5a. When the original texture is used for a damaged parcel, texture mapping is not trivial and we need to extrapolate the texture image. This extrapolation is done using pixel-wise nearest neighbor averaging and an exemplary result can be seen in Fig. 6.5b. In addition, we randomly add to each texture

- 0-3 logos from the Large Logo Dataset (LLD) [Sag+17]
- 1 shipping label from a mix of 30 labels from [Dör+19] and 65 labels found online, with a probability of 0.6
- 0-2 fragile labels from 16 labels found online, with a probability of 0.4

An example for a final cardboard texture with labels and logos is visualized in Fig. 6.5c.

6.2.4 Rendering Details

We sample 200 models randomly for each of the 66 groups, yielding more than $13\,000$ scenes, which we render with 1080×720 resolution. Damaged models and cuboid-shaped models, respectively, are sampled with a probability of 50% and textures are generated as described before. We add 0-3 randomly sampled distractor models from $\mathbb{G}_{\mathrm{D}}$ to the scene and use environment maps from Gardner et al. [Gar+17] for realistic scene contexts. We permit an occlusion of up to 30%

[1] See https://blendermarket.com/products/cardboard [Last accessed on Sept. 20, 2024].

(a) Cardboard **(b)** Extrapolation **(c)** Labels & Logos

Figure 6.5: Examples for generated textures: (a) Plain cardboard texture, (b) extrapolation of existing textures for damaged parcels and (c) cardboard texture with labels and logos. [©2023 IEEE]

of the model of interest and generate a new image composition if the criteria is not met.

All assets that were used follow a $0.7, 0.15, 0.15$ split between training, validation and test data. These splits were respected in the generation of the rendered images. To have realistic poses of the objects we restrict the elevation angle to lie between $20°$ and $60°$ degrees. The azimuth angle is sampled freely for intact and between $-30°$ and $30°$ degrees for deformed models, such that the damage is visible and not self-occluded. We add small random rotations to the *lookat* configuration resulting from azimuth and elevation angle and vary the focal length slightly at random.

6.3 Approach [Nau+23a]

We present our novel model architecture CubeRefine R-CNN that is targeted towards reconstructing potentially deformed cuboid objects such as parcels in Section 6.3.1. Furthermore, we present details on our training procedure in Section 6.3.2.

6.3.1 Neural Network Architecture

Our model CubeRefine R-CNN extends Cube R-CNN [Bra+23] by adding an iterative mesh refinement (cf. Fig. 6.1). Cube R-CNN is a general architecture that combines 2D detection with 3D bounding box estimation. Its architecture consist of a backbone network for feature extraction, which is followed by a Region Proposal Network (RPN) [Ren+17]. We follow the original work and use a DLA-34-FPN [Yu+18; Lin+17] as backbone. The generated region proposals are then passed on to two different branches. The first branch is a *Box Head*, which outputs a 2D bounding box and the category label. The second branch estimates the 3D bounding box and is called *Cube Head.* It takes 7×7 feature maps pooled from the region-aligned backbone features and passes them to two fully connected layers with hidden dimension 1024. A final fully connected layer predicts 13 parameters which represent the 3D bounding box. Note, that this architecture could be easily extended to encompass a full Mask R-CNN [He+17] by adding segmentation. For details, we refer to Brazil et al. [Bra+23].

For the mesh refinement, we extend the *Cube Head* by subdividing its 8-point mesh triangulation output four times to obtain an initial mesh prediction of sufficient granularity. Note, that without the iterative subdivision, the mesh representation would be too coarse to accurately represent parcel deformations. The subdivided mesh is then passed on to the mesh refinement stage. We follow Gkioxari et al. [GMJ19], and use three refinement stages with three graph convolutions each. In each stage, image features from the backbone are aligned with the vertices of the current mesh version and graph convolutions are applied to compute a positional offset for each vertex in the mesh. These mesh offsets should morph the current mesh representation such that the mesh closely depicts the real parcel shape. We experimented with different options for message passing within the graph such as Residual Gated Graph Convolution [BL18], EG [Tai+22] and GATv2 [BAY22]. Since no significant improvements were observed, we stick to the original architecture.

CubeRefine R-CNN leverages a cuboid prior, which is a valid assumption for both cuboid-shaped and most damaged parcels. Compared to Mesh R-CNN, the *Cube*

Head is more lightweight than the *Voxel Head*. Moreover, our model predicts both, the original shape of the parcel and the possibly deformed current shape of the parcel at the same time. We discuss the advantages of this in more detail in Section 6.4.3.

6.3.2 Training Procedure

We follow the same training procedure for all our training runs. We choose a batch size of 16, use Stochastic Gradient Descent with Momentum (SGD+M) with a base learning rate of 0.02. The learning rate increases linearly from 0.002 over the first 1500 iterations. Subsequently, we divide the learning rate by four in iterations 7500, $12\,500$ and $17\,500$. The maximum number of iterations is set to $20\,000$.

During our experiments, we consider two different backbones, namely a ResNet-50 [He+16] and a DLA-34 [Yu+18], both in combination with a Feature Pyramid Network (FPN) [Lin+17]. We freeze the backbone weights at stage four and initialize them using pre-trained weights from Gkioxari et al. [GMJ19] and Brazil et al. [Bra+23].

6.4 Evaluation” [Nau+23a]

In the following, we present our evaluation of 2D bounding box detection, 3D bounding box detection and shape reconstruction on synthetic and real data. Due to the lack of annotated real data of damaged parcels, the quantitative real-world evaluation only presents results on cuboid-shaped parcels. We benchmark our model against Pix2Mesh [Wan+18][2], Mesh R-CNN [GMJ19] and Cube R-CNN [Bra+23] by training and evaluating on the respective splits of Parcel3D. Unless

[2] We use the implementation of Gkioxari et al. [GMJ19].

stated otherwise, we use the same DLA-34-FPN backbone and three mesh refinement stages with three graph convolutions each, to enable a direct comparison between approaches. We present results for the original version of Mesh R-CNN with a ResNet-50-FPN backbone, however, focus on the comparable results in the following.

All results are summarized in Table 6.1 and Table 6.2, and we present details on the evaluation for synthetic data in Section 6.4.1 and for real data in Section 6.4.2. Finally, we summarize the findings focusing on the real-world applicability in Section 6.4.3.

6.4.1 Synthetic Data

We consider the case of intact parcels and damaged parcels separately by evaluating only on the respective subsets of the Parcel3D test dataset. The performance for 2D bounding box detection is very high for all models on our presented synthetic dataset Parcel3D with the lowest observed Box AP being 92.1 (cf. Table 6.1).

Considering 3D bounding box detection in the case of cuboid-shaped parcels, Cube R-CNN and CubeRefine R-CNN perform best w.r.t. Mesh AP_{75}, Chamfer Distance and Normal Consistency, since they explicitly model cuboid-shaped objects. Our additional mesh refinement increases performance compared to the base model Cube R-CNN by 2.8 percentage points in Mesh AP_{75}. Mesh R-CNN still performs competitively, and the qualitative inspection (cf. Fig. 6.6) suggests that differences mainly stem from difficulties in reconstructing the nonvisible, (self-)occluded parts of objects. Cube R-CNN and CubeRefine R-CNN do not suffer from this problem as much, since symmetry is imposed by the predicted 3D bounding box.

Model	Dataset	Box AP	Box AP_{50}	Box AP_{75}	Mesh AP_{50}	Mesh AP_{75}	Chamfer Distance (↓)	Normal Consistency
Pix2Mesh [Wan+18]	Intact	96.0 (0.5)	98.9 (0.5)	98.7 (0.2)	89.6 (1.0)	48.5 (1.6)	0.311 (0.086)	0.901 (0.001)
Mesh R-CNN (RN50) [GMJ19]	Intact	93.2 (0.4)	98.3 (0.3)	97.9 (0.3)	82.9 (1.2)	42.6 (1.1)	1.924 (1.214)	0.886 (0.001)
Mesh R-CNN [GMJ19]	Intact	95.9 (0.5)	98.9 (0.5)	98.7 (0.2)	**92.9 (1.6)**	67.0 (2.8)	0.225 (0.088)	0.914 (0.001)
Cube R-CNN [Bra+23]	Intact	97.1 (0.1)	**99.0 (0.0)**	**99.0 (0.0)**	92.0 (0.3)	74.4 (2.0)	0.159 (0.016)	0.925 (0.001)
CubeRefine R-CNN (ours)	Intact	**97.1 (0.0)**	**99.0 (0.0)**	**99.0 (0.0)**	92.8 (0.2)	**77.2 (1.2)**	**0.128 (0.002)**	**0.929 (0.001)**
Pix2Mesh [Wan+18]	Damaged	95.1 (0.6)	**99.8 (0.1)**	98.8 (0.2)	84.3 (1.2)	12.4 (1.4)	0.750 (0.553)	0.866 (0.002)
Mesh R-CNN (RN50) [GMJ19]	Damaged	92.1 (0.4)	99.6 (0.1)	98.9 (0.4)	78.8 (0.7)	9.0 (0.4)	0.599 (0.322)	0.859 (0.001)
Mesh R-CNN [GMJ19]	Damaged	94.6 (0.5)	99.2 (0.5)	98.8 (0.3)	**91.1 (0.5)**	**26.1 (1.9)**	0.860 (0.436)	**0.880 (0.002)**
Cube R-CNN [Bra+23]	Damaged	95.0 (0.2)	99.0 (0.0)	**99.0 (0.0)**	32.6 (0.5)	0.1 (0.0)	0.494 (0.004)	0.806 (0.000)
CubeRefine R-CNN (ours)	Damaged	**95.2 (0.1)**	99.0 (0.0)	**99.0 (0.0)**	70.7 (0.7)	4.1 (0.2)	**0.293 (0.003)**	0.861 (0.000)
Pix2Mesh [Wan+18]	Real	74.4 (1.9)	93.4 (1.7)	89.3 (2.4)	27.8 (2.1)	2.3 (0.6)	2.112 (0.060)	0.744 (0.006)
Mesh R-CNN (RN50) [GMJ19]	Real	**82.1 (0.7)**	**99.0 (0.0)**	**97.8 (0.1)**	32.0 (0.4)	5.0 (1.0)	1.965 (0.050)	0.756 (0.002)
Mesh R-CNN [GMJ19]	Real	70.6 (5.0)	89.2 (5.9)	84.4 (5.7)	29.4 (2.7)	4.9 (1.5)	2.153 (0.073)	0.742 (0.008)
Cube R-CNN [Bra+23]	Real	43.4 (6.9)	52.8 (8.3)	49.9 (7.3)	30.1 (5.8)	**13.3 (4.3)**	0.875 (0.041)	0.808 (0.003)
CubeRefine R-CNN (ours)	Real	41.5 (5.8)	50.3 (6.6)	47.6 (6.5)	**32.3 (4.2)**	13.1 (3.0)	**0.814 (0.062)**	**0.828 (0.006)**

Table 6.1: Quantitative performance analysis of mesh reconstruction on different datasets. The Mesh AP is the mean area under the Precision-Recall curve for F1@0.3$> x$, as in [GMJ19]. We repeated all trainings five times and report mean values with standard deviations in parentheses. The best mean performance for each dataset type is highlighted.

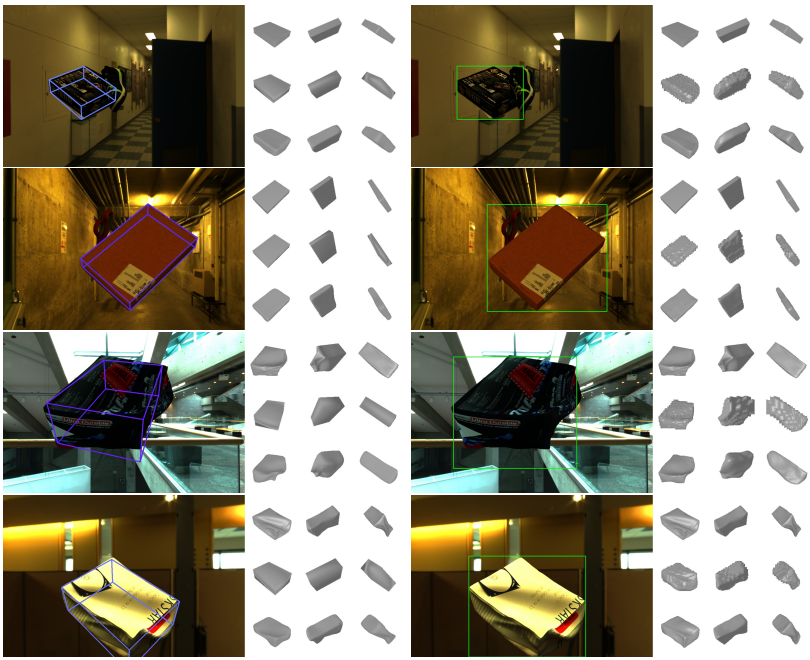
Model	Dataset	AP3D	$AP3D_{15}$	$AP3D_{25}$
Cube R-CNN [Bra+23]	Intact	69.5 (0.8)	81.6 (1.1)	74.4 (1.4)
CubeRefine R-CNN (ours)	Intact	69.3 (0.6)	80.9 (0.5)	74.1 (1.1)
Cube R-CNN [Bra+23]	Damaged	86.6 (0.3)	94.4 (0.6)	89.9 (0.8)
CubeRefine R-CNN (ours)	Damaged	86.5 (0.6)	94.6 (0.6)	89.7 (0.6)
Cube R-CNN [Bra+23]	Real	53.3 (8.6)	53.8 (8.7)	53.8 (8.7)
CubeRefine R-CNN (ours)	Real	50.6 (6.8)	51.1 (6.8)	51.1 (6.8)

Table 6.2: Quantitative performance analysis of 3D object detection for Cube R-CNN and CubeRefine R-CNN on different datasets. The average precision for 3D IoU (AP3D) is computed as in [Bra+23]. We repeated all trainings five times and report mean values with standard deviations in parentheses.

Considering only damaged parcels, we observe that predicting a voxel occupancy grid as done in Mesh R-CNN is advantageous. Mesh R-CNN performs best in Mesh AP and Normal Consistency. Despite high-quality 3D object detection, as suggested by the results in Table 6.2, CubeRefine R-CNN has difficulties to adopt to the fine-grained meshes of damaged parcels. This is observed in the considerably lower Mesh AP. However, the better Chamfer Distance suggests that general alignment with the ground truth is very high for CubeRefine R-CNN. This can also be observed in qualitative samples as visualized in Fig. 6.6 and might be caused by the symmetry the 3D bounding box imposes for (self-)occluded object parts. Cube R-CNN performs poorly, as it only predicts 3D bounding boxes and thus, cannot take the damages into account.

6.4.2 Real Data

For the evaluation of the usability of our approach in real-world applications, we use a dataset of parcels photos in various environments [Nau+22]. The dataset was generated using a custom camera rig to capture images with a depth and a stereo camera at the same time. The depth information is then used



(a) CubeRefine R-CNN (ours) **(b)** Mesh R-CNN [GMJ19]

Figure 6.6: Exemplary qualitative results for synthetic intact (row 1, 2) and damaged parcels (row 3, 4) for (a) CubeRefine R-CNN and (b) Mesh R-CNN. Per model, the input image with the detected 2D or 3D bounding box is shown on the left, and a 3×3 grid of mesh reconstructions on the right. Each column of the grid shows a different viewing angle, and the rows contain ground truth, 3D bounding box or voxelization (depending on the model) and refined mesh, respectively. [©2023 IEEE]

to automatically generate annotations, which can be projected onto the stereo images. The validation dataset comprises 96 and the test dataset 297 images. Note, that it contains only normal parcels, since the annotation generation process was automated using the assumption of a cuboid shape.

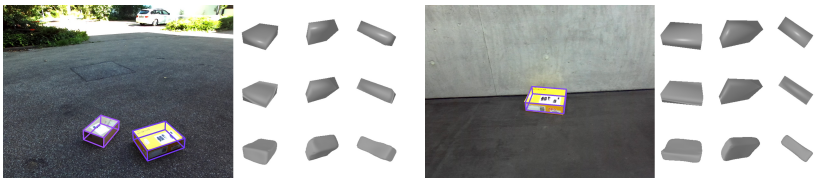
Shape reconstruction on real images of cuboid-shaped parcels is more challenging due to the reality gap, as can be seen from the generally lower performance in Table 6.1. CubeRefine R-CNN performs best despite having a low 2D bounding box detection precision compared to Mesh R-CNN. Note, that Mesh AP, Chamfer Distance and Normal Consistency were computed on meshes normalized within a unit cube, due to the scale ambiguity. While Cube R-CNN is able to estimate scale, our synthetic training data is generated randomly, and thus, does not allow a scale transfer to the real world.

We present qualitative samples in Fig. 6.7 and observe accurate reconstructions, when the object is localized correctly. However, common error cases include not being able to distinguish nearby positioned parcels and inaccurate or missing localizations (cf. Fig. 6.7b). Since there are no real-world datasets with full 3D annotations, we focus on brief insights into our qualitative inspection of damaged parcels. The simulated deformation process that was presented in Section 6.2.2 does not seem to represent the great variance of real-world deformations closely enough. Thus, performance on real-world data is still limited as can be seen in Fig. 6.7c.

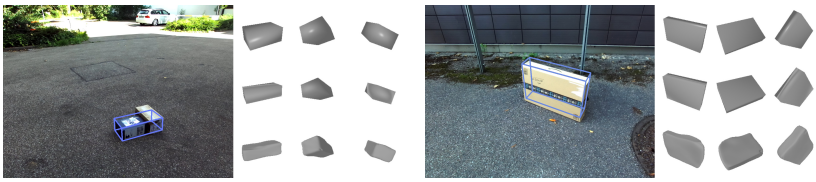
6.4.3 Applicability Summary

We summarize the advantages and limitations of our approach, and present brief insights into using damage quantification and tampering detection in practice.

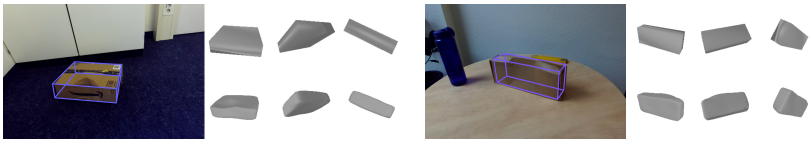
Advantages. We argue, that while Mesh R-CNN performs best in the case of damaged parcels, our approach is still advantageous for real-world application due to the following reasons: (1) our approach is more lightweight and predicts both



(a) Successful Reconstructions



(b) Problematic Cases



(c) Damaged Parcels

Figure 6.7: Exemplary qualitative results for real parcels using CubeRefine R-CNN. We show the input image with the projected 3D bounding box on the left, and a 3×3 grid of mesh reconstructions on the right. Each column shows a different viewing angle, and the rows contain ground truth, 3D bounding box and refined mesh, respectively. Note, that for damaged parcels no ground truth is available. [©2023 IEEE]

the current, potentially deformed shape of an object and its original shape at the same time. This allows a direct 3D mesh comparison between the original and the deformed shape for damage quantification. (2) The lower Mesh AP and better Chamfer Distance compared to Mesh R-CNN suggest that our model represents the overall damage pattern well, however, is not as detailed as Mesh R-CNN. We argue, that this is sufficient for damage pattern recognition in 3D, which is only enabled by our model. (3) 3D bounding box detection enables using viewpoint invariant parcel side surface representations for tampering detection, as will be explained in the respective paragraph.

Limitations. While CubeRefine R-CNN has important advantages for real-world use-cases, enabling reliable deployment in real-world scenarios is still challenging, presumably due to the constrained variance of deformations within Parcel3D and the domain shift caused by our training on synthetic data. Furthermore, it is important to note that we focus on deformations of the packaging and do not treat other types of damages which frequently occur in practice (e.g. water damage). It is also not possible to reliably infer the impact of packaging deformations on the state of the transported good. This information is essential to estimate economic damages.

Damage Quantification. To utilize our model for automated deformation quantification and pattern recognition, metrics for 3D mesh comparison are necessary. The change in volume between the original and current shape constitutes a simple metric that can be readily computed and interpreted. However, mere volume analysis does not take the deformation location into account. To remedy this, extending the axis-aligned pointcloud representation of the original 3D model by the per-point distance to the nearest neighbor of its potentially deformed version, and clustering in this 4D space can help to identify areas that underwent the strongest deformations. Further clustering across parcel instances can provide insights into damage patterns. Moreover, normalized voxel grid occupancy differences can be analyzed by considering the union of the voxelized meshes and subtracting their intersection.

Tampering Detection. From the 3D bounding box output of CubeRefine R-CNN we can infer the visible parcel side surfaces and project them back onto the image. For each such parcel side surface, a perspective transformation can be applied to obtain normalized fronto-parallel views. These representations have already been successfully used for tampering detection [NZO18] and re-identification [Rui+20]. For tampering detection, recent advances in change detection [Shi+20] could be leveraged.

7 Discussion

In the following, we summarize the conclusions from our work in Section 7.1 and present an outlook for future research ideas in Section 7.2.

7.1 Conclusion

In this work, we tackled the problem of vision-based damage and tampering assessment for parcels in transportation logistics and warehousing. After motivating the importance of process automation in logistics using computer vision with a special focus on damage and tampering assessment in Chapter 1, we introduced the necessary fundamentals in the area of computer vision in Chapter 2. Subsequently, we presented a detailed literature review on computer vision applications in logistics based on [Nau+23b] in Chapter 3. From this review the necessity for novel approaches for damage and tampering assessment that only rely on a single RGB image as input was deducted.

In Chapter 4, we treated the problem of robust parcel localization and segmentation that is based on our prior publication [Nau+22]. This serves as the foundation to tackle downstream tasks such as keypoint detection and 3D reconstruction, which are crucial for damage and tampering assessment. We extended the dataset generation pipeline of Dwibedi et al. [DMH17] by image scraping and selection to enable fully automated and flexible instance segmentation dataset generation. To analyze the effect of different image selection strategies, we presented a case study for parcels. All evaluations were performed on our newly collected real-world dataset of cuboid-shaped parcels with full 2D and 3D annotations. Results show

that the detection accuracy for bounding boxes and segmentation masks is high, as indicated by the Box AP of 68.5 and the Mask AP of 82.4. Furthermore, we found that manually selecting relevant instances from the scraped image pool is not superior to simple automated post-processing in the considered case study.

Tampering detection for parcels was analyzed in-depth in Chapter 5 based on our prior work [Nau+24]. We focused on three different tampering types (tape, writing, label), and proposed a pipeline that combines keypoint and change detection. The knowledge of the eight parcel corner points is exploited to generate viewpoint invariant parcel side surface representations by applying perspective transformations. These representations alleviate difficulties arising from the different viewing angles of the images and enable the detection of appearance changes per parcel side surface. To tackle change detection, we combined different image homogenization approaches with image similarity metrics. Image homogenization reduces the impact of lighting differences while the image similarity metrics determine whether a parcel side surface has been tampered with or not by thresholding. Our approach reached 81% accuracy and an F1-Score of 0.83, when combining SimSaC [Par+22] with the Learned Perceptual Image Patch Similarity (LPIPS) [Zha+18].

Based on our prior work [Nau+23a], we presented an approach for damage assessment of parcels in Chapter 6. We extended CubeRefine R-CNN [Bra+23] by an iterative mesh refinement [Wan+18; GMJ19] to leverage a cuboid prior while at the same time being able to adapt to deformations. An added advantage of this is that our approach estimates the current, potentially deformed shape as well as the prior cuboid shape of the pristine parcel. This enables detailed damage assessment and quantification by comparing full 3D meshes. Additionally, we presented Parcel3D, a novel dataset of synthetic images of damaged and intact parcels with full 2D and 3D annotations. CubeRefine R-CNN achieves competitive performance in terms of Mesh AP, however, reliable deployment in real-world scenarios remains challenging due to the large diversity of potential parcel deformations.

We evaluated all our approaches on real-world test data. This allows us to quantify the suitability of the approaches for real-world applications when there is no domain gap between our test data and a specific industry use-case. Since in practice, a domain gap will likely be present (e.g. new tampering types or damage patterns), it is crucial to perform a quantitative analysis on the data distribution of the specific industrial application, in order to obtain performance indicators that help to assess the business case. Moreover, it is important to comment on the different error types and their importance for practical applications, since common machine learning metrics are frequently not well suited to analyze business cases. For example, if the object detection from Chapter 4 is used for counting items to automatically invoice them, it is crucial to get the number of items exactly correct. At the same time, if our approach for damage detection from Chapter 6 would be used for inferring cargo volumes, overshooting the total volume would be less problematic than underestimating it. Underestimating volume would mean that the subsequent truck load planning will overestimate the capacity and shipping for some items might have to be postponed. Regarding the tampering detection approach from Chapter 5, many false positives can cause high workloads for manual cargo revision. The occurring costs for manual inspection might exceed the potential savings from identifying tampering in a timely and automated manner, thus, rendering the business case impractical. Thus, it is evident that to make business decisions that involve the machine learning approaches presented in this work, it is essential to take into account potential domain gaps and to determine and quantify the most relevant performance measures from the business perspective.

7.2 Future Work

There are several promising research directions to further exploit advances in computer vision for applications in transportation logistics and warehousing.

The presented approach on tampering detection can be extended by utilizing state-of-the-art keypoint detection approaches and by incorporating prior geometric

information, such as the vanishing point loss [Rui+20] or 2D/3D correspondences [Li+20]. Damage assessment would strongly benefit from the availability of real-world datasets of damaged parcels with full 2D and 3D annotations. This would allow a more in-depth performance analysis for real-world use-cases and help to bridge the domain shift from synthetic to real data. Another extension would be tackling the limitations of our work. For tampering detection, the development of a robust parcel re-identification module that relies on reading out labels while simultaneously taking visual cues into account seems promising. Regarding damage detection, the consideration of ruptures and water damage would be interesting.

While the focus of this work was on simple sensor setups, novel approaches using multisensory setups and potentially constrained environments are important to enable a reliable and continuous monitoring along the supply chain. Especially congested conveyor belts with numerous overlapping parcels and plastic mailers are a challenging industrial use-case, which has not been tackled yet by the literature. The retrieval of freight volumes while simultaneously checking for damage and tampering without the necessity for human intervention would also be an interesting topic for future research.

Resource Overview

We provide several online resources for our publications, such as project websites, permissively licensed code and publicly available datasets. Per chapter, we list the relevant papers including their resources in the following.

Chapter 3

Alexander Naumann et al. *Literature Review: Computer Vision Applications in Transportation Logistics and Warehousing*. Preprint. Apr. 2023. arXiv: 2304.06009. URL: https://arxiv.org/abs/2304.06009 (Last accessed on Sept. 20, 2024) [Nau+23b]

- *Website:* https://a-nau.github.io/cv-in-logistics
- *Code:* https://github.com/a-nau/cv-in-logistics

Chapter 4

Alexander Naumann et al. "Scrape, Cut, Paste and Learn: Automated Dataset Generation Applied to Parcel Logistics". In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Nassau, Bahamas, Dec. 2022, pp. 1026–1031. DOI: 10.1109/ICMLA55696.2022.00171 [Nau+22]

- *Website:* https://a-nau.github.io/parcel2d
- *Code:*

  - *Image scraping:* https://github.com/a-nau/easy-image-scraping
  - *Dataset generation:* https://github.com/a-nau/synthetic-dataset-generation
  - *CNN training:* https://github.com/a-nau/image-selection-and-cnn-training

- *Dataset:* https://zenodo.org/record/8031971

Chapter 5

Alexander Naumann et al. "TAMPAR: Visual Tampering Detection for Parcel Logistics in Postal Supply Chains". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Waikoloa, Hawaii, USA, Jan. 2024, pp. 8076–8086 [Nau+24]

- *Website:* https://a-nau.github.io/tampar
- *Code:* https://github.com/a-nau/tampar
- *Dataset:* https://zenodo.org/records/10057090

Chapter 6

Alexander Naumann et al. "Parcel3D: Shape Reconstruction from Single RGB Images for Applications in Transportation Logistics". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Vancouver, Canada, June 2023, pp. 4402–4412. DOI: 10.1109/cvprw59228.2023.00463 [Nau+23a]

- *Website:* https://a-nau.github.io/parcel3d
- *Code:* https://github.com/a-nau/CubeRefine-R-CNN
- *Dataset:* https://zenodo.org/record/8032204

Bibliography

[Arn+19] Eduardo Arnold, Omar Y. Al-Jarrah, Mehrdad Dianati, Saber Fallah, David Oxtoby, and Alex Mouzakitis. "A Survey on 3D Object Detection Methods for Autonomous Driving Applications". In: *IEEE Transactions on Intelligent Transportation Systems* 20.10 (Oct. 2019), pp. 3782–3795. ISSN: 1558-0016. DOI: 10.1109/TITS.2019.2892405.

[Arp+20] Pierluigi Arpenti, Riccardo Caccavale, Gianmarco Paduano, Giuseppe Andrea Fontanelli, Vincenzo Lippiello, Luigi Villani, and Bruno Siciliano. "RGB-D Recognition and Localization of Cases for Robotic Depalletizing in Supermarkets". In: *IEEE Robotics and Automation Letters* 5.4 (Oct. 2020), pp. 6233–6238. ISSN: 2377-3766, 2377-3774. DOI: 10/gh547b.

[Bai+20] Qiang Bai, Shaobo Li, Jing Yang, Qisong Song, Zhiang Li, and Xingxing Zhang. "Object Detection Recognition and Robot Grasping Based on Machine Learning: A Survey". In: *IEEE Access* 8 (2020), pp. 181855–181879. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3028740.

[Bal13] Roberto Baldwin. *Shipshape: Tracking 40 Years of FedEx Tech*. Apr. 2013. URL: https://www.wired.com/2013/04/40-years-of-fedex/ (Last accessed on Sept. 20, 2024).

[BAM19] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. "Multimodal Machine Learning: A Survey and Taxonomy". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*

41.2 (Feb. 2019), pp. 423–443. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2018.2798607.

[Bay+08] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. "Speeded-Up Robust Features (SURF)". In: *Computer Vision and Image Understanding*. Similarity Matching in Computer Vision and Multimedia 110.3 (June 2008), pp. 346–359. ISSN: 1077-3142. DOI: 10/ffsc9r.

[BAY22] Shaked Brody, Uri Alon, and Eran Yahav. "How Attentive Are Graph Attention Networks?" In: *Proceedings of the International Conference on Learning Representations*. Virtual, Jan. 2022. URL: https://openreview.net/forum?id=F72ximsx7C1 (Last accessed on Sept. 20, 2024).

[BBS21] Robert Brylka, Benjamin Bierwirth, and Ulrich Schwanecke. "AI-based Recognition of Dangerous Goods Labels and Metric Package Features". In: *Proceedings of the Hamburg International Conference of Logistics (HICL)*. Hamburg, Germany, Dec. 2021, pp. 245–272. ISBN: 978-3-7549-2770-0. DOI: 10/gpc9mv.

[Bel+20] Saifullahi Aminu Bello, Shangshu Yu, Cheng Wang, Jibril Muhmmad Adam, and Jonathan Li. "Review: Deep Learning on 3D Point Clouds". In: *Remote Sensing* 12.11 (Jan. 2020), p. 1729. ISSN: 2072-4292. DOI: 10.3390/rs12111729.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. First. Information Science and Statistics. New York: Springer, 2006. ISBN: 978-0-387-31073-2.

[BK00] A. Bicchi and V. Kumar. "Robotic Grasping and Contact: A Review". In: *Proceedings of the IEEE International Conference on Robotics and Automation*. Vol. 1. San Francisco, CA, USA, Apr. 2000, pp. 348–353. DOI: 10.1109/ROBOT.2000.844081.

[BL18] Xavier Bresson and Thomas Laurent. *Residual Gated Graph ConvNets*. Preprint. Apr. 2018. arXiv: 1711.07553. URL: https://arxiv.org/abs/1711.07553 (Last accessed on Sept. 20, 2024).

[Bor+13] Hagen Borstell, Liu Cao, Jewgeni Kluth, and Klaus Richter. “Prozessintegrierte Volumenerfassung von logistischen Palettenstrukturen auf Basis von Low-Cost-Tiefenbildsensoren”. In: *Tagungsband / 3D-NordOst, Anwendungsbezogener Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D-Daten, im Rahmen der GFaI-Workshop-Familie NordOst*. Berlin, Germany, Dec. 2013, pp. 115–124. ISBN: 978-3-942709-09-5.

[Bor+14] Hagen Borstell, Jewgeni Kluth, Marcel Jaeschke, Cathrin Plate, Bernd Gebert, and Klaus Richter. “Pallet Monitoring System Based on a Heterogeneous Sensor Network for Transparent Warehouse Processes”. In: *Proceedings of the Sensor Data Fusion: Trends, Solutions, Applications (SDF)*. Bonn, Germany, Oct. 2014, pp. 1–6. DOI: 10.1109/SDF.2014.6954718.

[Bor21] Hagen Borstell. “Bildbasierte Zustandserfassung in der Logistik”. Doctoral dissertation. Magdeburg: Otto-von-Guericke-Universität Magdeburg, 2021. DOI: 10.13140/RG.2.2.28823.09121.

[Bra+23] Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. “Omni3D: A Large Benchmark and Model for 3D Object Detection in the Wild”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. Vancouver, Canada, June 2023. DOI: 10.1109/cvpr52729.2023.01264.

[Bro+93] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. “Signature Verification Using a "Siamese" Time Delay Neural Network”. In: *Proceedings of the International Conference on Neural Information Processing Systems*. San Francisco, CA, USA, Nov. 1993, pp. 737–744. DOI: 10.1142/9789812797926_0003.

[BSB20] Robert Brylka, Ulrich Schwanecke, and Benjamin Bierwirth. “Camera Based Barcode Localization and Decoding in Real-World Applications”. In: *Proceedings of the International Conference on Omnilayer Intelligent Systems (COINS)*. Barcelona, Spain, Aug. 2020, pp. 1–8. DOI: 10.1109/COINS49042.2020.9191416.

[BTL19] Salma Benslimane, Simon Tamayo, and Arnaud de La Fortelle. *Classifying Logistic Vehicles in Cities Using Deep Learning*. Preprint. June 2019. arXiv: 1906.11895. URL: https://arxiv.org/abs/1906.11895 (Last accessed on Sept. 20, 2024).

[Bu+17] Yanling Bu, Lei Xie, Jia Liu, Bingbing He, Yinyin Gong, and Sanglu Lu. “3-Dimensional Reconstruction on Tagged Packages via RFID Systems”. In: *Proceedings of the Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. San Diego, CA, USA, June 2017, pp. 1–9. ISBN: 978-1-5090-6599-8. DOI: 10.1109/SAHCN.2017.7964911.

[BUE16] Marco Bonini, Augusto Urru, and Wolfgang Echelmeyer. “Fast Deployable Autonomous Systems for Order Picking - How Small and Medium Size Enterprises Can Benefit from the Automation of the Picking Process”. In: *Proceedings of the International Conference on Informatics in Control, Automation and Robotics*. Lisbon, Portugal, 2016, pp. 479–484. ISBN: 978-989-758-198-4. DOI: 10.5220/0005997804790484.

[Can86] John Canny. “A Computational Approach to Edge Detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6 (Nov. 1986), pp. 679–698. ISSN: 1939-3539. DOI: 10/fn3fdk.

[CCW19] Con Cronin, Andrew Conway, and Joseph Walsh. “State-of-the-Art Review of Autonomous Intelligent Vehicles (AIV) Technologies for the Automotive and Manufacturing Industry”. In: *Proceedings of the Irish Signals and Systems Conference (ISSC)*. Derry, Ireland, June 2019, pp. 1–6. DOI: 10/ggfhnc.

[Cha+15] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. *ShapeNet: An Information-Rich 3D Model Repository*. Preprint. Dec. 2015. arXiv: 1512.03012. URL: https://arxiv.org/abs/1512.03012 (Last accessed on Sept. 20, 2024).

[Che+22] Haoming Chen, Runyang Feng, Sifan Wu, Hao Xu, Fengcheng Zhou, and Zhenguang Liu. "2D Human Pose Estimation: A Survey". In: *Multimedia Systems* 29.5 (Nov. 2022), pp. 3115–3138. ISSN: 1432-1882. DOI: 10.1007/s00530-022-01019-0.

[Chi+20] Davide Chiaravalli, Gianluca Palli, Riccardo Monica, Jacopo Aleotti, and Dario Lodi Rizzini. "Integration of a Multi-Camera Vision System and Admittance Control for Robotic Industrial Depalletizing". In: *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. Vol. 1. Vienna, Austria, Sept. 2020, pp. 667–674. DOI: 10.1109/ETFA46521.2020.9212020.

[Cho+16] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. "3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction". In: *Computer Vision – ECCV 2016*. Cham, Oct. 2016, pp. 628–644. ISBN: 978-3-319-46484-8. DOI: 10.1007/978-3-319-46484-8_38.

[Cla+19] Sascha Clausen, Claudius Zelenka, Tobias Schwede, and Reinhard Koch. "Parcel Tracking by Detection in Large Camera Networks". In: *Proceedings of the German Conference on Pattern Recognition*. Lecture Notes in Computer Science. Cham, 2019, pp. 89–104. ISBN: 978-3-030-12939-2. DOI: 10/gpc9md.

[Cor+18] Nikolaus Correll, Kostas E. Bekris, Dmitry Berenson, Oliver Brock, Albert Causo, Kris Hauser, Kei Okada, Alberto Rodriguez, Joseph M. Romano, and Peter R. Wurman. "Analysis and Observations From the First Amazon Picking Challenge". In: *IEEE Transactions on Automation Science and Engineering* 15.1 (Jan. 2018), pp. 172–188. ISSN: 1558-3783. DOI: 10.1109/TASE.2016.2600527.

[CTH20] Yucheng Chen, Yingli Tian, and Mingyi He. “Monocular Human Pose Estimation: A Survey of Deep Learning-Based Methods”. In: *Computer Vision and Image Understanding* 192 (Mar. 2020). ISSN: 1077-3142. DOI: 10.1016/j.cviu.2019.102897.

[Das+22] Sagnik Das, Ke Ma, Zhixin Shu, and Dimitris Samaras. “Learning an Isometric Surface Parameterization for Texture Unwrapping”. In: *Computer Vision – ECCV*. Ed. by Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner. Vol. 13697. Cham: Springer Nature Switzerland, Oct. 2022, pp. 580–597. ISBN: 978-3-031-19835-9 978-3-031-19836-6. DOI: 10.1007/978-3-031-19836-6_33.

[Den+19] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Youssef Zidan, Dmitry Olefir, Mohamad Elbadrawy, Ahsan Lodhi, and Harinandan Katam. *BlenderProc*. Preprint. Oct. 2019. arXiv: 1911.01911. URL: https://arxiv.org/abs/1911.01911 (Last accessed on Sept. 20, 2024).

[DMH17] Debidatta Dwibedi, Ishan Misra, and Martial Hebert. “Cut, Paste and Learn: Surprisingly Easy Synthesis for Instance Detection”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy, Oct. 2017, pp. 1310–1319. ISBN: 978-1-5386-1032-9. DOI: 10.1109/ICCV.2017.146.

[Dör+19] Laura Dörr, Felix Brandt, Anne Meyer, and Martin Pouls. “Lean Training Data Generation for Planar Object Detection Models in Unsteady Logistics Contexts”. In: *Proceedings of the IEEE International Conference On Machine Learning And Applications (ICMLA)*. Boca Raton, FL, USA, Dec. 2019, pp. 329–334. ISBN: 978-1-72814-550-1. DOI: 10/ghd2cg.

[Dör+20a] Laura Dörr, Felix Brandt, Martin Pouls, and Alexander Naumann. “An Image Processing Pipeline for Automated Packaging Structure Recognition”. In: *Forum Bildverarbeitung*. Karlsruhe, Germany, 2020, pp. 239–251. ISBN: 978-3-7315-1053-6. DOI: 10.5445/KSP/1000124383.

[Dör+20b] Laura Dörr, Felix Brandt, Martin Pouls, and Alexander Naumann. "Fully-Automated Packaging Structure Recognition in Logistics Environments". In: *Proceedings of the International Conference on Emerging Technologies and Factory Automation*. Vienna, Austria, Sept. 2020. ISBN: 978-1-72818-956-7. DOI: 10.1109/ETFA46521.2020.9212152.

[Dör+21] Laura Dörr, Felix Brandt, Alexander Naumann, and Martin Pouls. "TetraPackNet: Four-Corner-Based Object Detection in Logistics Use-Cases". In: *Proceedings of the DAGM German Conference on Pattern Recognition*. Bonn, Germany, 2021. ISBN: 978-3-030-92659-5. DOI: 10.1007/978-3-030-92659-5_35.

[Dör+23] Laura Dörr, Katharina Glock, Felix Brandt, Alexander Naumann, and Martin Pouls. "A Digital Measuring and Load Planning System for Large Transport Assets". In: *2023 International Scientific Symposium on Logistics: Conference Volume*. June 2023, pp. 49–55. DOI: 10.25366/2023.124.

[Dos+20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *Proceedings of the International Conference on Learning Representations*. Virtual, Oct. 2020. URL: https://openreview.net/forum?id=YicbFdNTTy (Last accessed on Sept. 20, 2024).

[Dow+22] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. "Google Scanned Objects: A High-Quality Dataset of 3D Scanned Household Items". In: *Proceedings of the International Conference on Robotics and Automation (ICRA)*. Philadelphia, PA, USA, May 2022, pp. 2553–2560. DOI: 10.1109/ICRA46639.2022.9811809.

[DP22] Shivam Duggal and Deepak Pathak. “Topologically-Aware Deformation Fields for Single-View 3D Reconstruction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA, June 2022, pp. 1526–1536. ISBN: 978-1-66546-946-3. DOI: 10.1109/CVPR52688.2022.00159.

[DT05] N. Dalal and B. Triggs. “Histograms of Oriented Gradients for Human Detection”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 1. San Diego, CA, USA, June 2005, pp. 886–893. DOI: 10/fjwckz.

[Du+21] Guoguang Du, Kai Wang, Shiguo Lian, and Kaiyong Zhao. “Vision-Based Robotic Grasping from Object Localization, Object Pose Estimation to Grasp Estimation for Parallel Grippers: A Review”. In: *Artificial Intelligence Review* 54.3 (Mar. 2021), pp. 1677–1734. ISSN: 1573-7462. DOI: 10.1007/s10462-020-09888-5.

[Dwi+16] Debidatta Dwibedi, Tomasz Malisiewicz, Vijay Badrinarayanan, and Andrew Rabinovich. *Deep Cuboid Detection: Beyond 2D Bounding Boxes*. Preprint. Nov. 2016. arXiv: 1611.10010. URL: https://arxiv.org/abs/1611.10010 (Last accessed on Sept. 20, 2024).

[FB87] Martin A. Fischler and Robert C. Bolles. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: *Readings in Computer Vision*. Ed. by Martin A. Fischler and Oscar Firschein. San Francisco, CA, USA: Morgan Kaufmann, 1987, pp. 726–740. ISBN: 978-0-08-051581-6. DOI: 10.1016/B978-0-08-051581-6.50070-2.

[Fen+21] Hao Feng, Yuechen Wang, Wengang Zhou, Jiajun Deng, and Houqiang Li. “DocTr: Document Image Transformer for Geometric Unwarping and Illumination Correction”. In: *Proceedings of the 29th ACM International Conference on Multimedia*. New York, NY, USA: Association for Computing Machinery, Oct. 2021, pp. 273–281. ISBN: 978-1-4503-8651-7. DOI: 10.1145/3474085.3475388.

[FH04] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. “Efficient Graph-Based Image Segmentation”. In: *International Journal of Computer Vision* 59.2 (Sept. 2004), pp. 167–181. ISSN: 0920-5691. DOI: 10/fdmw8q.

[Fot+21] Johannes Fottner, Dana Clauer, Fabian Hormes, Michael Freitag, Thies Beinke, Ludger Overmeyer, Simon Nicolas Gottwald, Ralf Elbert, Tessa Sarnow, Thorsten Schmidt, et al. “Autonomous Systems in Intralogistics – State of the Art and Future Research Challenges”. In: *Logistics Research* 14.2 (Feb. 2021). DOI: 10.23773/2021_2.

[Fra+21] Giuseppe Fragapane, René de Koster, Fabio Sgarbossa, and Jan Ola Strandhagen. “Planning and Control of Autonomous Mobile Robots for Intralogistics: Literature Review and Research Agenda”. In: *European Journal of Operational Research* 294.2 (Oct. 2021), pp. 405–426. ISSN: 03772217. DOI: 10.1016/j.ejor.2021.01.019.

[FS95] Yoav Freund and Robert E. Schapire. “A Desicion-Theoretic Generalization of on-Line Learning and an Application to Boosting”. In: *Proceedings of the Computational Learning Theory*. Lecture Notes in Computer Science. Berlin, Heidelberg, 1995, pp. 23–37. ISBN: 978-3-540-49195-8. DOI: 10.1007/3-540-59119-2_166.

[FSG17] Haoqiang Fan, Hao Su, and Leonidas Guibas. “A Point Set Generation Network for 3D Object Reconstruction from a Single Image”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA, July 2017, pp. 2463–2471. ISBN: 978-1-5386-0457-1. DOI: 10.1109/CVPR.2017.264.

[Fu+21] Kui Fu, Jiansheng Peng, Qiwen He, and Hanxiao Zhang. “Single Image 3D Object Reconstruction Based on Deep Learning: A Review”. In: *Multimedia Tools and Applications* 80.1 (Jan. 2021), pp. 463–498. ISSN: 1573-7721. DOI: 10/gjmj5p.

[Fur+18] F. Furrer, T. Novkovic, M. Fehr, A. Gawel, M. Grinvald, T. Sattler, R. Siegwart, and J. Nieto. “Incremental Object Database: Building 3D Models from Multiple Partial Observations”. In: *Proceedings of*

the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Madrid, Spain, Oct. 2018, pp. 6835–6842. DOI: 10.1109/iros.2018.8594391.

[FZL19] Jiaojiao Fang, Lingtao Zhou, and Guizhong Liu. *3D Bounding Box Estimation for Autonomous Vehicles by Cascaded Geometric Constraints and Depurated 2D Detections Using 3D Results*. Preprint. Sept. 2019. arXiv: 1909.01867. URL: https://arxiv.org/abs/1909.01867 (Last accessed on Sept. 20, 2024).

[FZL21] Ernesto Fontana, William Zarotti, and Dario Lodi Rizzini. "A Comparative Assessment of Parcel Box Detection Algorithms for Industrial Applications". In: *Proceedings of the European Conference on Mobile Robots (ECMR)*. Bonn, Germany, Aug. 2021, pp. 1–6. DOI: 10/gpc9mm.

[Gar+17] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. "Learning to Predict Indoor Illumination from a Single Image". In: *ACM Transactions on Graphics* 36.6 (Nov. 2017), pp. 1–14. ISSN: 0730-0301. DOI: 10/gcqfdn.

[Gar+98] G. Garibotto, S. Masciangelo, P. Bassino, C. Coelho, A. Pavan, and M. Marson. "Industrial Exploitation of Computer Vision in Logistic Automation: Autonomous Control of an Intelligent Forklift Truck". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Vol. 2. Leuven, Belgium, 1998, pp. 1459–1464. ISBN: 978-0-7803-4300-9. DOI: 10.1109/ROBOT.1998.677310.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. Adaptive Computation and Machine Learning Series. Cambridge, MA, USA: MIT Press, 2016. ISBN: 978-0-262-33737-3.

[Ge+21] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. *YOLOX: Exceeding YOLO Series in 2021*. Preprint. Aug. 2021.

arXiv: 2107.08430. URL: https://arxiv.org/abs/2107.08430 (Last accessed on Sept. 20, 2024).

[Ghi+21] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. "Simple Copy-Paste Is a Strong Data Augmentation Method for Instance Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA, June 2021, pp. 2917–2927. ISBN: 978-1-66544-509-2. DOI: 10.1109/CVPR46437.2021.00294.

[Glo+20] Holger Glockner, Kai Jannek, Johannes Mahn, and Björn Theis. *Augmented Reality in Logistics: Changing the Way We See Logistics - A DHL Perspective*. DHL Customer Solutions & Innovation. 2020. URL: https://www.dhl.com/discover/content/dam/dhl/downloads/interim/full/dhl-csi-augmented-reality-report.pdf (Last accessed on Sept. 20, 2024).

[GMJ19] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. "Mesh R-CNN". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Seoul, Korea, 2019, pp. 9784–9794. ISBN: 978-1-7281-4803-8. DOI: 10.1109/iccv.2019.00988.

[GMR17] E. Grilli, F. Menna, and F. Remondino. "A Review of Point Clouds Segmentation and Classification Algorithms". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)* XLII-2/W3 (Feb. 2017), pp. 339–344. ISSN: 2194-9034. DOI: 10.5194/isprs-archives-XLII-2-W3-339-2017.

[Goo+14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative Adversarial Nets". In: *Proceedings of the Advances in Neural Information Processing Systems*. Vol. 2. Montréal, Canada, 2014, pp. 2672–2680. ISBN: 9781510800410. URL: https://dl.acm.org/doi/abs/10.5555/2969033.2969125 (Last accessed on Sept. 20, 2024).

[Gri+19] Margarita Grinvald, Fadri Furrer, Tonci Novkovic, Jen Jen Chung, Cesar Cadena, Roland Siegwart, and Juan Nieto. “Volumetric Instance-Aware Semantic Mapping and 3D Object Discovery”. In: *IEEE Robotics and Automation Letters* 4.3 (July 2019), pp. 3037–3044. ISSN: 2377-3766. DOI: 10.1109/LRA.2019.2923960.

[Gri+23] Daniel Grimm, Maximilian Zipfl, Felix Hertlein, Alexander Naumann, Jürgen Lüttin, Steffen Thoma, Stefan Schmid, Lavdim Halilaj, Achim Rettinger, and J. Marius Zöllner. “Heterogeneous Graph-based Trajectory Prediction Using Local Map Context and Social Interactions”. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC)*. Bilbao, Spain, Sept. 2023.

[Grz+16] Réné Grzeszick, Sascha Feldhorst, Christian Mosblech, Gernot A. Fink, and Michael Ten Hompel. “Camera-Assisted Pick-by-feel”. In: *Logistics Journal: Proceedings* 2016 (2016). ISSN: 2192-9084. DOI: 10.2195/lj_proc_grzeszick_en_201610_01.

[GWM18] Matheus Gadelha, Rui Wang, and Subhransu Maji. “Multiresolution Tree Networks for 3D Point Cloud Processing”. In: *Computer Vision - ECCV 2018*. Ed. by Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss. Cham: Springer International Publishing, 2018, pp. 103–118. ISBN: 978-3-030-01224-3. DOI: 10.1007/978-3-030-01234-2_7.

[Ham20] William L. Hamilton. *Graph Representation Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Cham: Springer International Publishing, 2020. ISBN: 978-3-031-00460-5 978-3-031-01588-5. DOI: 10.1007/978-3-031-01588-5.

[HBC16] Doug Haanpaa, Glenn Beach, and Charles J. Cohen. “Machine Vision Algorithms for Robust Pallet Engagement and Stacking”. In: *Proceedings of the IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*. Washington, DC, USA, Oct. 2016, pp. 1–8. ISBN: 978-1-5090-3284-6. DOI: 10.1109/AIPR.2016.8010590.

[He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA, June 2016, pp. 770–778. ISBN: 978-1-4673-8851-1. DOI: 10/gdcfkn.

[He+17] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. “Mask R-CNN”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy, Oct. 2017, pp. 2980–2988. ISBN: 978-1-5386-1032-9. DOI: 10.1109/ICCV.2017.322.

[Her+21] David M. Herold, Marek Ćwiklicki, Kamila Pilch, and Jasmin Mikl. “The Emergence and Adoption of Digitalization in the Logistics and Supply Chain Industry: An Institutional Perspective”. In: *Journal of Enterprise Information Management* 34.6 (Nov. 2021), pp. 1917–1938. ISSN: 1741-0398. DOI: 10.1108/JEIM-09-2020-0382.

[HLB21] Xian-Feng Han, Hamid Laga, and Mohammed Bennamoun. “Image-Based 3D Object Reconstruction: State-of-the-Art and Trends in the Deep Learning Era”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.5 (May 2021), pp. 1578–1604. ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: 10/ggth3k.

[HM17] Marian Himstedt and Erik Maehle. “Online Semantic Mapping of Logistic Environments Using RGB-D Cameras”. In: *International Journal of Advanced Robotic Systems* 14.4 (July 2017). DOI: 10.1177/1729881417720781.

[HM18] J. Hinxlage and J. Möller. *Ladungsträgerzahlung per Smartphone*. Jahresbericht Fraunhofer IML. 2018. URL: https://www.iml.fraunhofer.de/content/dam/iml/de/documents/OE%20983/Presse/Jahresberichte/Jahresbericht_2018_Fraunhofer_IML.pdf (Last accessed on Sept. 20, 2024).

[HN23] Felix Hertlein and Alexander Naumann. “Template-Guided Illumination Correction for Document Images with Imperfect Geometric

Reconstruction". In: *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. Paris, France, 2023, pp. 904–913. DOI: 10.1109/ICCVW60793.2023.00097.

[HNP23] Felix Hertlein, Alexander Naumann, and Patrick Philipp. "Inv3D: A High-Resolution 3D Invoice Dataset for Template-Guided Single-Image Document Unwarping". In: *International Journal on Document Analysis and Recognition (IJDAR)* (Apr. 2023). ISSN: 1433-2825. DOI: 10.1007/s10032-023-00434-x.

[Hoc+16] Maximilian Hochstein, Johannes Glöckle, Thomas Meyer, and Kai Furmans. "Packassistent - Assistenzsystem für die Qualitätskontrolle während des Packprozesses". In: *Logistics Journal: Proceedings* 2016 (2016). DOI: 10.2195/lj_proc_hochstein_de_201610_01.

[Hou62] Paul V. C. Hough. "Method and Means for Recognizing Complex Patterns". US3069654A. US Patent. Dec. 1962.

[Hu+21a] Haohao Hu, Fabian Immel, Johannes Janosovits, Martin Lauer, and Christoph Stiller. "A Cuboid Detection and Tracking System Using A Multi RGBD Camera Setup for Intelligent Manipulation and Logistics". In: *Proceedings of the IEEE International Conference on Automation Science and Engineering (CASE)*. Lyon, France, Aug. 2021, pp. 1097–1103. DOI: 10/gpfnh2.

[Hu+21b] Tao Hu, Liwei Wang, Xiaogang Xu, Shu Liu, and Jiaya Jia. "Self-Supervised 3D Mesh Reconstruction from Single Images". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA, June 2021, pp. 5998–6007. ISBN: 978-1-66544-509-2. DOI: 10.1109/cvpr46437.2021.00594.

[ID18] Eldar Insafutdinov and Alexey Dosovitskiy. "Unsupervised Learning of Shape and Pose with Differentiable Point Clouds". In: *Proceedings of the Conference on Neural Information Processing Systems (NeuRIPS)*. Montreal, Canada, 2018, pp. 2807–2817. URL:

https://dl.acm.org/doi/10.5555/3327144.3327204 (Last accessed on Sept. 20, 2024).

[Jac01] Paul Jaccard. “Distribution de La Flore Alpine Dans Le Bassin Des Dranses et Dans Quelques Régions Voisines”. In: *Bull Soc Vaudoise Sci Nat* 37 (1901), pp. 241–272. DOI: 10.5169/seals-266440.

[Jia+22] Xiangwei Jiang, Rujiao Long, Nan Xue, Zhibo Yang, Cong Yao, and Gui-Song Xia. “Revisiting Document Image Dewarping by Grid Regularization”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA, June 2022, pp. 4533–4542. ISBN: 978-1-66546-946-3. DOI: 10.1109/CVPR52688.2022.00450.

[JN12] Gao Jie and Liu Ning. “An Improved Adaptive Threshold Canny Edge Detection Algorithm”. In: *Proceedings of the International Conference on Computer Science and Electronics Engineering*. Vol. 1. Hangzhou, China, Mar. 2012, pp. 164–168. DOI: 10/ggkhf3.

[Joc+22] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, Github Account NanoCode012, Yonghye Kwon, Kalen Michael, TaoXie, Jiacong Fang, imyhxy, et al. *Ultralytics/Yolov5: V7.0 - YOLOv5 SOTA Realtime Instance Segmentation*. Zenodo. Nov. 2022. DOI: 10.5281/zenodo.7347926.

[KA05] Hüseyin N. Karaca and Cüneyt Akınlar. “A Multi-camera Vision System for Real-Time Tracking of Parcels Moving on a Conveyor Belt”. In: *Proceedings of the Computer and Information Sciences (ISCIS)*. Ed. by pInar Yolum, Tunga Güngör, Fikret Gürgen, and Can Özturan. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2005, pp. 708–717. ISBN: 978-3-540-32085-2. DOI: 10.1007/11569596_73.

[Kal60] Rudolph Emil Kalman. “A New Approach to Linear Filtering and Prediction Problems”. In: *Transactions of the ASME-Journal of Basic Engineering* 82.Series D (Mar. 1960), pp. 35–45. DOI: 10.1115/1.3662552.

[Kam+22] Teerawat Kamnardsiri, Phasit Charoenkwan, Chommaphat Malang, and Ratapol Wudhikarn. “1D Barcode Detection: Novel Benchmark Datasets and Comprehensive Comparison of Deep Convolutional Neural Network Approaches”. In: *Sensors* 22.22 (Nov. 2022). ISSN: 1424-8220. DOI: 10.3390/s22228788.

[Kan+18] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. “Learning Category-Specific Mesh Reconstruction from Image Collections”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Ed. by Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss. Vol. 11219. Cham: Springer International Publishing, 2018, pp. 386–402. ISBN: 978-3-030-01266-3 978-3-030-01267-0. DOI: 10.1007/978-3-030-01267-0_23.

[Kha+22] Muhammad Saif Ullah Khan, Alain Pagani, Marcus Liwicki, Didier Stricker, and Muhammad Zeshan Afzal. “Three-Dimensional Reconstruction from a Single RGB Image Using Deep Learning: A Review”. In: *Journal of Imaging* 8.9 (Sept. 2022), p. 225. ISSN: 2313-433X. DOI: 10.3390/jimaging8090225.

[Kle+20] Kilian Kleeberger, Richard Bormann, Werner Kraus, and Marco F. Huber. “A Survey on Learning-Based Robotic Grasping”. In: *Current Robotics Reports* 1.4 (Dec. 2020), pp. 239–249. ISSN: 2662-4087. DOI: 10.1007/s43154-020-00021-6.

[Klü+22] Simon Klüttermann, Jérôme Rutinowski, Christopher Reining, Moritz Roidl, and Emmanuel Müller. “Towards Graph Representation Based Re-Identification of Chipwood Pallet Blocks”. In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Nassau, Bahamas, Dec. 2022, pp. 1543–1550. DOI: 10.1109/ICMLA55696.2022.00279.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Communications of the ACM* 60.6 (May 2012), pp. 84–90. ISSN: 00010782. DOI: 10.1145/3065386.

[KTS18] Surajkumar Goraksha Kumbhar, Rachana B. Thombare, and Amitkumar B. Salunkhe. “Automated Guided Vehicles for Small Manufacturing Enterprises: A Review”. In: *SAE International Journal of Materials and Manufacturing* 11.3 (Sept. 2018), pp. 253–258. ISSN: 1946-3987. DOI: 10/ggfhm9.

[KU19] Sultan Daud Khan and Habib Ullah. “A Survey of Advances in Vision-Based Vehicle Re-Identification”. In: *Computer Vision and Image Understanding* 182 (May 2019), pp. 50–63. ISSN: 1077-3142. DOI: 10.1016/j.cviu.2019.03.001.

[Kuc+19] Haluk Kucuk, Mohammad T. Al Muallim, Fikret Yılmaz, and Metin Kahraman. “Development of a Dimensions Measurement System Based on Depth Camera for Logistic Applications”. In: *Proceedings of the International Conference on Machine Vision (ICMV)*. Munich, Germany, Mar. 2019, p. 93. ISBN: 978-1-5106-2748-2 978-1-5106-2749-9. DOI: 10.1117/12.2523123.

[Küc13] Markus Kückelhaus. *DHL - Low-cost Sensor Technology*. DHL Customer Solutions & Innovation. 2013. URL: https://pdf4pro.com/amp/view/low-cost-sensor-technology-dhl-express-83245.html (Last accessed on Sept. 20, 2024).

[Kum+22] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. “DEVIANT: Depth EquiVarIAnt NeTwork for Monocular 3D Object Detection”. In: *Computer Vision – ECCV*. Ed. by Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner. Vol. 13669. Cham: Springer Nature Switzerland, 2022, pp. 664–683. ISBN: 978-3-031-20076-2 978-3-031-20077-9. DOI: 10.1007/978-3-031-20077-9_39.

[Kuz+20] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. “The Open Images Dataset V4: Unified Image Classification, Object Detection, and Visual Relationship Detection at Scale”. In: *International Journal of Computer*

Vision 128.7 (July 2020), pp. 1956–1981. ISSN: 0920-5691, 1573-1405. DOI: 10/ghf8dc.

[KW16] Thomas N. Kipf and Max Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *Proceedings of the International Conference on Learning Representations*. San Juan, Puerto Rico, Nov. 2016. URL: https://openreview.net/forum?id=SJU4ayYgl (Last accessed on Sept. 20, 2024).

[LD18] Hei Law and Jia Deng. "CornerNet: Detecting Objects as Paired Keypoints". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Munich, Germany, 2018, pp. 734–750. DOI: 10.1007/s11263-019-01204-1.

[LH17] Ilya Loshchilov and Frank Hutter. "SGDR: Stochastic Gradient Descent with Warm Restarts". In: *Proceedings of the International Conference on Learning Representations*. Toulon, France, Apr. 2017. URL: https://openreview.net/forum?id=Skq89Scxx (Last accessed on Sept. 20, 2024).

[Li+12] Xingyan Li, Ian Yen-Hung Chen, Stephen Thomas, and Bruce A MacDonald. "Using Kinect for Monitoring Warehouse Order Picking Operations". In: *Proceedings of Australasian Conference on Robotics and Automation*. Wellington, New Zealand, Dec. 2012, p. 7. URL: https://www.araa.asn.au/acra/acra2012/papers/pap108.pdf (Last accessed on Sept. 20, 2024).

[Li+20] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. "RTM3D: Real-Time Monocular 3D Detection from Object Keypoints for Autonomous Driving". In: *Computer Vision - ECCV 2020*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12348. Cham: Springer International Publishing, 2020, pp. 644–660. ISBN: 978-3-030-58579-2 978-3-030-58580-8. DOI: 10.1007/978-3-030-58580-8_38.

[Li+21] Yifan Li, Yingchun Niu, Yang Liu, Li Zheng, Zichen Wang, and Wenming Zhe. “Computer Vision Based Conveyor Belt Congestion Recognition in Logistics Industrial Parks”. In: *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. Västerås, Sweden, Sept. 2021, pp. 1–8. DOI: 10/gn8sk6.

[Li+22] Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, and Furu Wei. “DiT: Self-supervised Pre-training for Document Image Transformer”. In: *Proceedings of the 30th ACM International Conference on Multimedia*. Lisbon, Portugal, 2022, pp. 3530–3539. DOI: 10.1145/3503161.3547911.

[Lin+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. “Microsoft COCO: Common Objects in Context”. In: *Proceedings of the Computer Vision - ECCV 2014*. Ed. by David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 740–755. ISBN: 978-3-319-10602-1. DOI: 10.1007/978-3-319-10602-1_48.

[Lin+17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. “Feature Pyramid Networks for Object Detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, Hawaii, USA, July 2017, pp. 936–944. DOI: 10/gc7rk2.

[Liu+19] Chen Liu, Kihwan Kim, Jinwei Gu, Yasutaka Furukawa, and Jan Kautz. “PlaneRCNN: 3D Plane Detection and Reconstruction From a Single Image”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA, June 2019, pp. 4445–4454. ISBN: 978-1-72813-293-8. DOI: 10/ggkg6t.

[Low99] D.G. Lowe. “Object Recognition from Local Scale-Invariant Features”. In: *Proceedings of the IEEE International Conference on*

Computer Vision. Vol. 2. Kerkyra, Greece, Sept. 1999, 1150–1157 vol.2. ISBN: 0-7695-0164-8. DOI: 10/dqfs59.

[LWP13] Suraphol Laotrakunchai, Akarapas Wongkaew, and Karn Patanukhom. "Measurement of Size and Distance of Objects Using Mobile Devices". In: *Proceedings of the International Conference on Signal-Image Technology Internet-Based Systems*. Kyoto, Japan, Dec. 2013, pp. 156–161. DOI: 10/ggdhmq.

[LYL21] Yuxuan Liu, Yuan Yixuan, and Ming Liu. "Ground-Aware Monocular 3D Object Detection for Autonomous Driving". In: *IEEE Robotics and Automation Letters* 6.2 (2021), pp. 919–926. DOI: 10.1109/LRA.2021.3052442.

[Ma+18] Ke Ma, Zhixin Shu, Xue Bai, Jue Wang, and Dimitris Samaras. "DocUNet: Document Image Unwarping via a Stacked U-Net". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, Utah, USA, 2018, pp. 4700–4709. DOI: 10.1109/cvpr.2018.00494.

[Ma+23] Xinzhu Ma, Wanli Ouyang, Andrea Simonelli, and Elisa Ricci. "3D Object Detection from Images for Autonomous Driving: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Dec. 2023), pp. 1–20. DOI: 10.1109/TPAMI.2023.3346386.

[Mal+21] M. I. Malyshev, S. A. Braginsky, E. Yu. Faddeeva, and S. S. Gogolin. "Artificial Neural Network Detection of Damaged Goods by Packaging State". In: *Proceedings of the Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED)*. Moscow, Russia, Nov. 2021, pp. 1–7. DOI: 10/gpc9mx.

[Mar+20] Amir Markovitz, Inbal Lavi, Or Perel, Shai Mazor, and Roee Litman. "Can You Read Me Now? Content Aware Rectification Using Angle Supervision". In: *Proceedings of the Computer Vision – ECCV*. Lecture Notes in Computer Science. Cham, 2020, pp. 208–223. ISBN: 978-3-030-58610-2. DOI: 10.1007/978-3-030-58610-2_13.

[Mät+16] Benedikt Mättig, Isabel Lorimer, Jana Jost, and Thomas Kirks. "Untersuchung des Einsatzes von Augmented Reality im Verpackungsprozess unter Berücksichtigung spezifischer Anforderungen an die Informationsdarstellung sowie die ergonomische Einbindung des Menschen in den Prozess". In: *Logistics Journal: Proceedings* 2016.10 (2016). DOI: 10.2195/lj_proc_maettig_de_201610_01.

[May+20] Christopher Mayershofer, Dimitrij-Marian Holm, Benjamin Molter, and Johannes Fottner. "LOCO: Logistics Objects in Context". In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Miami, Florida, 2020, pp. 612–617. DOI: 10/gn8st9.

[Mem+20] Jamshed Memon, Maira Sami, Rizwan Ahmed Khan, and Mueen Uddin. "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)". In: *IEEE Access* 8 (2020), pp. 142642–142668. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3012542.

[Men+21] Thomas Mensink, Jasper Uijlings, Alina Kuznetsova, Michael Gygli, and Vittorio Ferrari. "Factors of Influence for Transfer Learning across Diverse Appearance Domains and Task Types". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.12 (2021), pp. 9298–9314. ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: 10.1109/TPAMI.2021.3129870.

[Mes+19] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. "Occupancy Networks: Learning 3D Reconstruction in Function Space". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA, June 2019. DOI: 10/ghgqdg.

[Mey92] F. Meyer. "Color Image Segmentation". In: *Proceedings of the International Conference on Image Processing and Its Applications*. Maastricht, Netherlands, Apr. 1992, pp. 303–306.

[MF18] B. Molter and J. Fottner. “Real-Time Pallet Localization with 3D Camera Technology for Forklifts in Logistic Environments”. In: *Proceedings of the IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*. Singapore, July 2018, pp. 297–302. DOI: 10.1109/soli.2018.8476740.

[MF19] Benjamin Molter and Johannes Fottner. “Semi-Automatic Pallet Pick-up as an Advanced Driver Assistance System for Forklifts”. In: *Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC)*. Auckland, New Zealand, Oct. 2019, pp. 4464–4469. DOI: 10.1109/ITSC.2019.8917189.

[MGF20] Christopher Mayershofer, Tao Ge, and Johannes Fottner. “Towards Fully-Synthetic Training for Industrial Applications”. In: *Proceedings of the International Conference on Logistics, Informatics and Service Sciences (LISS)*. 2020, pp. 765–782. ISBN: 978-981-33-4359-7. DOI: 10.1007/978-981-33-4359-7_53.

[MHH19] Carina Mieth, Philipp Humbeck, and Georg Herzwurm. “A Survey on the Potentials of Indoor Localization Systems in Production”. In: *Advances in Production, Logistics and Traffic*. Ed. by Uwe Clausen, Sven Langkau, and Felix Kreuz. Cham: Springer International Publishing, 2019, pp. 142–154. ISBN: 978-3-030-13534-8 978-3-030-13535-5. DOI: 10.1007/978-3-030-13535-5_11.

[Mih+15] Rãzvan-George Mihalyi, Kaustubh Pathak, Narunas Vaskevicius, Tobias Fromm, and Andreas Birk. “Robust 3D Object Modeling with a Low-Cost RGBD-sensor and AR-markers for Applications with Untrained End-Users”. In: *Robotics and Autonomous Systems* 66 (Apr. 2015), pp. 1–17. ISSN: 0921-8890. DOI: 10.1016/j.robot.2015.01.005.

[Mil+20] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. “NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis”. In: *Proceedings of the Computer Vision - ECCV 2020*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm.

Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, pp. 405–421. ISBN: 978-3-030-58452-8. DOI: 10.1007/978-3-030-58452-8_24.

[Min+22] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. "Image Segmentation Using Deep Learning: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.7 (July 2022), pp. 3523–3542. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3059968.

[Mis+19] Vaishali Mishra, Harsh K Kapadia, Tanish H Zaveri, and Bhanu Prasad Pinnamaneni. "Development of Low-Cost Embedded Vision System with a Case Study on 1D Barcode Detection". In: *Proceedings of Information and Communication Technology for Intelligent Systems (ICTIS)*. Vol. 1. 2019, pp. 505–513. DOI: 10.1007/978-981-13-1742-2_50.

[Moh+20] Ihab S. Mohamed, Alessio Capitanelli, Fulvio Mastrogiovanni, Stefano Rovetta, and Renato Zaccaria. "Detection, Localisation and Tracking of Pallets Using Machine Learning Techniques and 2D Range Data". In: *Neural Computing and Applications* 32.13 (July 2020), pp. 8811–8828. ISSN: 1433-3058. DOI: 10.1007/s00521-019-04352-0.

[Mor78] Jorge J. Moré. "The Levenberg-Marquardt Algorithm: Implementation and Theory". In: *Proceedings of the Numerical Analysis*. Lecture Notes in Mathematics. Berlin, Heidelberg, 1978, pp. 105–116. ISBN: 978-3-540-35972-2. DOI: 10/fv5mkj.

[Mul+22] Norman Muller, Andrea Simonelli, Lorenzo Porzi, Samuel Rota Bulo, Matthias Niessner, and Peter Kontschieder. "AutoRF: Learning 3D Object Radiance Fields from Single View Observations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA, June 2022, pp. 3961–3970. ISBN: 978-1-66546-946-3. DOI: 10.1109/CVPR52688.2022.00394.

[MV19] Stephan Müller and Felix Voigtländer. “Automated Trucks in Road Freight Logistics: The User Perspective”. In: *Advances in Production, Logistics and Traffic*. Ed. by Uwe Clausen, Sven Langkau, and Felix Kreuz. Cham: Springer International Publishing, 2019, pp. 102–115. ISBN: 978-3-030-13534-8 978-3-030-13535-5. DOI: 10.1007/978-3-030-13535-5_8.

[Nau+20] Alexander Naumann, Laura Dörr, Niels Ole Salscheider, and Kai Furmans. “Refined Plane Segmentation for Cuboid-Shaped Objects by Leveraging Edge Detection”. In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Miami, Florida, USA, Dec. 2020, pp. 432–437. DOI: 10.1109/ICMLA51294.2020.00096.

[Nau+22] Alexander Naumann, Felix Hertlein, Benchun Zhou, Laura Dorr, and Kai Furmans. “Scrape, Cut, Paste and Learn: Automated Dataset Generation Applied to Parcel Logistics”. In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Nassau, Bahamas, Dec. 2022, pp. 1026–1031. DOI: 10.1109/ICMLA55696.2022.00171.

[Nau+23a] Alexander Naumann, Felix Hertlein, Laura Dörr, and Kai Furmans. “Parcel3D: Shape Reconstruction from Single RGB Images for Applications in Transportation Logistics”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Vancouver, Canada, June 2023, pp. 4402–4412. DOI: 10.1109/cvprw59228.2023.00463.

[Nau+23b] Alexander Naumann, Felix Hertlein, Laura Dörr, Steffen Thoma, and Kai Furmans. *Literature Review: Computer Vision Applications in Transportation Logistics and Warehousing*. Preprint. Apr. 2023. arXiv: 2304.06009. URL: https://arxiv.org/abs/2304.06009 (Last accessed on Sept. 20, 2024).

[Nau+23c] Alexander Naumann, Felix Hertlein, Daniel Grimm, Maximilian Zipfl, Steffen Thoma, Achim Rettinger, Lavdim Halilaj, Juergen Luettin, Stefan Schmid, and Holger Caesar. “Lanelet2 for nuScenes:

Enabling Spatial Semantic Relationships and Diverse Map-Based Anchor Paths". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Vancouver, BC, Canada, June 2023, pp. 3247–3256. DOI: 10.1109/CVPRW59228.2023.00327.

[Nau+24] Alexander Naumann, Felix Hertlein, Laura Dörr, and Kai Furmans. "TAMPAR: Visual Tampering Detection for Parcel Logistics in Postal Supply Chains". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Waikoloa, Hawaii, USA, Jan. 2024, pp. 8076–8086.

[Nav+20] K. L. Navaneet, Ansu Mathew, Shashank Kashyap, Wei-Chih Hung, Varun Jampani, and R. Venkatesh Babu. "From Image Collections to Point Clouds With Self-Supervised Shape and Pose Networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA, June 2020, pp. 1129–1137. ISBN: 978-1-72817-168-5. DOI: 10.1109/cvpr42600.2020.00121.

[Nik21] Sergey I. Nikolenko. *Synthetic Data for Deep Learning*. Vol. 174. Springer Optimization and Its Applications. Cham: Springer International Publishing, 2021. ISBN: 978-3-030-75177-7 978-3-030-75178-4. DOI: 10.1007/978-3-030-75178-4.

[Niu+21] Li Niu, Wenyan Cong, Liu Liu, Yan Hong, Bo Zhang, Jing Liang, and Liqing Zhang. *Making Images Real Again: A Comprehensive Survey on Deep Image Composition*. Preprint. June 2021. arXiv: 2106.14490. URL: https://arxiv.org/abs/2106.14490 (Last accessed on Sept. 20, 2024).

[NKS18] Alexander Naumann, Oliver Kolb, and Matteo Semplice. "On a Third Order CWENO Boundary Treatment with Application to Networks of Hyperbolic Conservation Laws". In: *Applied Mathematics and Computation* 325 (May 2018), pp. 252–270. ISSN: 0096-3003. DOI: 10/ggqbkm.

[NZO18] Nicoletta Noceti, Luca Zini, and Francesca Odone. “A Multi-Camera System for Damage and Tampering Detection in a Postal Security Framework”. In: *EURASIP Journal on Image and Video Processing* 2018.1 (Feb. 2018), p. 11. ISSN: 1687-5281. DOI: 10.1186/s13640-017-0242-x.

[ÖAN16] Çagdas Özgür, Cyril Alias, and Bernd Noche. “Comparing Sensor-Based and Camera-Based Approaches to Recognizing the Occupancy Status of the Load Handling Device of Forklift Trucks”. In: *Logistics Journal: Proceedings* 2016.05 (2016). DOI: 10.2195/lj_proc_oezguer_en_201605_01.

[Par+21] Jin-Man Park, Jae-Hyuk Jang, Sahng-Min Yoo, Sun-Kyung Lee, Ue-Hwan Kim, and Jong-Hwan Kim. “ChangeSim: Towards End-to-End Online Scene Change Detection in Industrial Indoor Environments”. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Prague, Czech Republic, Sept. 2021, pp. 8578–8585. DOI: 10.1109/IROS51168.2021.9636350.

[Par+22] Jin-Man Park, Ue-Hwan Kim, Seon-Hoon Lee, and Jong-Hwan Kim. “Dual Task Learning by Leveraging Both Dense Correspondence and Mis-Correspondence for Robust Change Detection With Imperfect Matches”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA, June 2022, pp. 13739–13749. ISBN: 978-1-66546-946-3. DOI: 10.1109/CVPR52688.2022.01338.

[Pav+19] Dmytro Pavlichenko, Germán Martín García, Seongyong Koo, and Sven Behnke. “KittingBot: A Mobile Manipulation Robot for Collaborative Kitting in Automotive Logistics”. In: *Intelligent Autonomous Systems 15*. Ed. by Marcus Strand, Rüdiger Dillmann, Emanuele Menegatti, and Stefano Ghidoni. Vol. 867. Cham: Springer International Publishing, 2019, pp. 849–864. ISBN: 978-3-030-01369-1 978-3-030-01370-7. DOI: 10.1007/978-3-030-01370-7_66.

[PGB03] Patrick Pérez, Michel Gangnet, and Andrew Blake. “Poisson Image Editing”. In: *Proceedings of the ACM SIGGRAPH 2003 Papers*. SIGGRAPH ’03. New York, NY, USA, July 2003, pp. 313–318. ISBN: 978-1-58113-709-5. DOI: 10.1145/1201775.882269.

[Pin+03] David Pinto, Andrew McCallum, Xing Wei, and W. Bruce Croft. “Table Extraction Using Conditional Random Fields”. In: *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*. New York, NY, USA, July 2003, pp. 235–242. ISBN: 978-1-58113-646-3. DOI: 10.1145/860435.860479.

[Pra+15] C. Prasse, J. Stenzel, A. Böckenkamp, B. Rudak, K. Lorenz, F. Weichert, H. Müller, and M. ten Hompel. “New Approaches for Singularization in Logistic Applications Using Low Cost 3D Sensors”. In: *Sensing Technology: Current Status and Future Trends IV*. Ed. by Alex Mason, Subhas Chandra Mukhopadhyay, and Krishanthi Padmarani Jayasundera. Smart Sensors, Measurement and Instrumentation. Cham: Springer International Publishing, 2015, pp. 191–215. ISBN: 978-3-319-12898-6. DOI: 10.1007/978-3-319-12898-6_10.

[Qin+20] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R. Zaiane, and Martin Jagersand. “U2-Net: Going Deeper with Nested U-structure for Salient Object Detection”. In: *Pattern Recognition* 106 (Oct. 2020), p. 107404. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2020.107404.

[Rei09] Rupert Reif. “Entwicklung und Evaluierung eines Augmented Reality unterstützten Kommissioniersystems”. Doctoral dissertation. Garching b. München, Germany: Lehrstuhl für Fördertechnik Materialfluß Logistik (fml), Techn. Univ. München, 2009. URL: https://mediatum.ub.tum.de/doc/683943/document.pdf (Last accessed on Sept. 20, 2024).

[Ren+16] Colin Rennie, Rahul Shome, Kostas E. Bekris, and Alberto F. De Souza. “A Dataset for Improved RGBD-Based Object Detection and Pose Estimation for Warehouse Pick-and-Place”. In: *IEEE Robotics*

and Automation Letters 1.2 (July 2016), pp. 1179–1185. ISSN: 2377-3766. DOI: 10.1109/LRA.2016.2532924.

[Ren+17] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (June 2017), pp. 1137–1149. ISSN: 0162-8828, 2160-9292. DOI: 10.1109/TPAMI.2016.2577031.

[RF18] Joseph Redmon and Ali Farhadi. *YOLOv3: An Incremental Improvement*. Preprint. Apr. 2018. arXiv: 1804.02767. URL: https://arxiv.org/abs/1804.02767 (Last accessed on Sept. 20, 2024).

[Ria+17] Amir Riad, Christian Sporer, Syed Saqib Bukhari, and Andreas Dengel. "Classification and Information Extraction for Complex and Nested Tabular Structures in Images". In: *Proceedings of the IAPR International Conference on Document Analysis and Recognition (ICDAR)*. Vol. 01. Kyoto, Japan, Nov. 2017, pp. 1156–1161. DOI: 10.1109/ICDAR.2017.191.

[Rie+19] Maik Riestock, Karl Fessel, Thomas Depner, and Hagen Borstell. "Survey of Depth Cameras for Process-integrated State Detection in Logistics". In: *Proceedings of the Smart SysTech 2019; European Conference on Smart Objects, Systems and Technologies*. Magdeburg, Germany, June 2019, pp. 1–6.

[Rom+22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. "High-Resolution Image Synthesis With Latent Diffusion Models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. New Orleans, Louisiana, USA, 2022, pp. 10684–10695. DOI: 10.1109/cvpr52688.2022.01042.

[Rui+20] Zeng Rui, Ge Zongyuan, Denman Simon, Sridharan Sridha, and Fookes Clinton. "Geometry-Constrained Car Recognition Using a 3D Perspective Network". In: *Proceedings of the AAAI Conference*

on Artificial Intelligence 34.01 (Apr. 2020), pp. 1161–1168. ISSN: 2374-3468. DOI: 10.1609/aaai.v34i01.5468.

[Rus+10] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu. "Fast 3D Recognition and Pose Using the Viewpoint Feature Histogram". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. Taipei, Taiwan, Oct. 2010, pp. 2155–2162. DOI: 10.1109/IROS.2010.5651280.

[Rut+21] Jérôme Rutinowski, Christian Pionzewski, Tim Chilla, Christopher Reining, and Michael ten Hompel. "Towards Re-Identification for Warehousing Entities - A Work-in-Progress Study". In: *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. Västerås, Sweden, Sept. 2021, pp. 1–4. DOI: 10.1109/ETFA45728.2021.9613250.

[Rut+22a] Jérôme Rutinowski, Christian Pionzewski, Tim Chilla, Christopher Reining, and Michael Ten Hompel. "Deep Learning Based Re-Identification of Wooden Euro-pallets". In: *Proceedings of the 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*. Nassau, Bahamas, Dec. 2022, pp. 113–117. DOI: 10.1109/ICMLA55696.2022.00023.

[Rut+22b] Jérôme Rutinowski, Hazem Youssef, Anas Gouda, Christopher Reining, and Moritz Roidl. "The Potential of Deep Learning Based Computer Vision in Warehousing Logistics". In: *Logistics Journal: Proceedings* 2022 (2022), Issue 18. DOI: 10.2195/LJ_PROC_RUTINOWSKI_EN_202211_01.

[Sab+18] Lorenzo Sabattini, Mika Aikio, Patric Beinschob, Markus Boehning, Elena Cardarelli, Valerio Digani, Annette Krengel, Massimiliano Magnani, Szilard Mandici, Fabio Oleari, et al. "The PAN-Robots Project: Advanced Automated Guided Vehicle Systems for Industrial Logistics". In: *IEEE Robotics Automation Magazine* 25.1 (Mar. 2018), pp. 55–64. ISSN: 1558-223X. DOI: 10.1109/MRA.2017.2700325.

[Sag+17] Alexander Sage, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. *LLD - Large Logo Dataset - Version 0.1*. 2017. URL: https://data.vision.ee.ethz.ch/cvl/lld (Last accessed on Sept. 20, 2024).

[Sag04] Mazen Saghir. "The Concept of Packaging Logistics". In: *Proceedings of the Fifteenth Annual World Conference on Production and Operations Management Society (POMS)*. Cancun, Mexiko, 2004, pp. 1–31. URL: https://www.pomsmeetings.org/ConfProceedings/002/POMS_CD/Browse%20This%20CD/PAPERS/002-0283.pdf (Last accessed on Sept. 20, 2024).

[Sam+09] Mehul P. Sampat, Zhou Wang, Shalini Gupta, Alan Conrad Bovik, and Mia K. Markey. "Complex Wavelet Structural Similarity: A New Image Similarity Index". In: *IEEE Transactions on Image Processing* 18.11 (Nov. 2009), pp. 2385–2401. ISSN: 1941-0042. DOI: 10.1109/TIP.2009.2025923.

[Sar+20] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. "SuperGlue: Learning Feature Matching With Graph Neural Networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA, June 2020, pp. 4937–4946. ISBN: 978-1-72817-168-5. DOI: 10.1109/CVPR42600.2020.00499.

[Sch+07] R Schnabel, R Wahl, R Wessel, and R Klein. *Shape Recognition in 3D Point Clouds*. Technical Report CG-2007/1. Bonn, Germany, 2007. URL: https://cg.cs.uni-bonn.de/backend/v1/files/publications/cg-2007-1.pdf (Last accessed on Sept. 20, 2024).

[Sch+17] Max Schwarz, Anton Milan, Christian Lenz, Aura Munoz, Arul Selvam Periyasamy, Michael Schreiber, Sebastian Schuller, and Sven Behnke. "NimbRo Picking: Versatile Part Handling for Warehouse Automation". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Singapore, Singapore, May 2017, pp. 3032–3039. ISBN: 978-1-5090-4633-1. DOI: 10.1109/ICRA.2017.7989348.

[SCH15] Manolis Savva, Angel X. Chang, and Pat Hanrahan. “Semantically-Enriched 3D Models for Common-Sense Knowledge”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Boston, MA, USA, June 2015, pp. 24–31. ISBN: 978-1-4673-6759-2. DOI: 10.1109/CVPRW.2015.7301289.

[Sch15] Jürgen Schmidhuber. “Deep Learning in Neural Networks: An Overview”. In: *Neural Networks* 61 (Jan. 2015), pp. 85–117. ISSN: 0893-6080. DOI: 10.1016/j.neunet.2014.09.003.

[Sev+22] Jaime Sevilla, Lennart Heim, Anson Ho, Tamay Besiroglu, Marius Hobbhahn, and Pablo Villalobos. “Compute Trends Across Three Eras of Machine Learning”. In: *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. Padua, Italy, July 2022, pp. 1–8. DOI: 10.1109/IJCNN55064.2022.9891914.

[She+12] Ravindra Shetty, Rebeca Cáceres, John Pastrana, and Luis Rabelo. “Optical Container Code Recognition and Its Impact on the Maritime Supply Chain”. In: *Proceedings of the Industrial and Systems Engineering Research Conference*. Orlando, Florida, USA, 2012. URL: https://www.researchgate.net/publication/290103375_Optical_container_code_recognition_and_its_impact_on_the_maritime_supply_chain (Last accessed on Sept. 20, 2024).

[Shi+20] Wenzhong Shi, Min Zhang, Rui Zhang, Shanxiong Chen, and Zhao Zhan. “Change Detection Based on Artificial Intelligence: State-of-the-Art and Challenges”. In: *Remote Sensing* 12.10 (Jan. 2020), p. 1688. ISSN: 2072-4292. DOI: 10.3390/rs12101688.

[Son+17a] Ngo Tung Son, Bui Ngoc Anh, Tran Quy Ban, and Tran Binh Duong. “A Method to Construct Automatic Object Bounding-Box Estimation System Using 3D Cameras”. In: *International Journal of Science and Research (IJSR)* 6.7 (July 2017), pp. 961–965. ISSN: 23197064. DOI: 10.21275/ART20175316.

[Son+17b] Shuran Song, Fisher Yu, Andy Zeng, Angel X. Chang, Manolis Savva, and Thomas Funkhouser. "Semantic Scene Completion from a Single Depth Image". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA, July 2017, pp. 190–198. ISBN: 978-1-5386-0457-1. DOI: 10.1109/CVPR.2017.28.

[SRS20] Xavier Soria, Edgar Riba, and Angel Sappa. "Dense Extreme Inception Network: Towards a Robust CNN Model for Edge Detection". In: *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*. Snowmass Village, CO, USA, Mar. 2020, pp. 1912–1921. ISBN: 978-1-72816-553-0. DOI: 10.1109/WACV45572.2020.9093290.

[ST14] Dmitry Shvarts and Mart Tamre. "Bulk Material Volume Estimation Method and System for Logistic Applications". In: *Proceedings of the International DAAAM Baltic Conference Industrial Engineering*. Tallinn, Estonia, Apr. 2014. URL: https://www.researchgate.net/publication/293100589_Bulk_material_volume_estimation_method_and_system_for_logistic_applications (Last accessed on Sept. 20, 2024).

[Sta+15] Alessandra Staglianò, Nicoletta Noceti, Alessandro Verri, and Francesca Odone. "Online Space-Variant Background Modeling With Sparse Coding". In: *IEEE Transactions on Image Processing* 24.8 (Aug. 2015), pp. 2415–2428. ISSN: 1941-0042. DOI: 10/ggdmng.

[Sto+17] Marie-Hélène Stoltz, Vaggelis Giannikas, Duncan McFarlane, James Strachan, Jumyung Um, and Rengarajan Srinivasan. "Augmented Reality in Warehouse Operations: Opportunities and Barriers". In: *IFAC-PapersOnLine* 50.1 (July 2017), pp. 12979–12984. ISSN: 24058963. DOI: 10.1016/j.ifacol.2017.08.1807.

[STU11] Bernd Scholz-Reiter, Hendrik Thamer, and Claudio Uriarte. "An Approach for 3D Object Recognition of Universal Goods". In: *International Journal of Computers* 5.2 (2011), pp. 218–225. ISSN:

1998-4308. URL: https://www.naun.org/main/NAUN/computers/19-892.pdf (Last accessed on Sept. 20, 2024).

[Sub+21] Nishant Subramani, Alexandre Matton, Malcolm Greaves, and Adrian Lam. *A Survey of Deep Learning Approaches for OCR and Document Understanding*. Preprint. Feb. 2021. arXiv: 2011.13534. URL: https://arxiv.org/abs/2011.13534 (Last accessed on Sept. 20, 2024).

[Suh+19] Sungho Suh, Haebom Lee, Yong Oh Lee, Paul Lukowicz, and Jongwoon Hwang. "Robust Shipping Label Recognition and Validation for Logistics by Using Deep Neural Networks". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Taipei, Taiwan, Sept. 2019, pp. 4509–4513. DOI: 10/gpfqf5.

[Sun+18] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. "Beyond Part Models: Person Retrieval with Refined Part Pooling (and A Strong Convolutional Baseline)". In: *Computer Vision - ECCV 2018*. Ed. by Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss. Vol. 11208. Cham: Springer International Publishing, 2018, pp. 501–518. ISBN: 978-3-030-01224-3 978-3-030-01225-0. DOI: 10.1007/978-3-030-01225-0_30.

[Sun+20] Y Sun, Z X Liu, M Li, Z T Zeng, Z X Zong, and C L Ji. "An Object Recognition and Volume Calculation Method Based on Yolov3 and Depth Vision". In: *Journal of Physics: Conference Series* 1684.1 (Nov. 2020), p. 012009. ISSN: 1742-6588, 1742-6596. DOI: 10/gpc9mq.

[Sze22] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Texts in Computer Science. Cham: Springer International Publishing, 2022. ISBN: 978-3-030-34371-2 978-3-030-34372-9. DOI: 10.1007/978-3-030-34372-9.

[Tai+22] Shyam A. Tailor, Felix Opolka, Pietro Lio, and Nicholas Donald Lane. “Do We Need Anisotropic Graph Neural Networks?” In: *Proceedings of the International Conference on Learning Representations*. Virtual, Jan. 2022. URL: https://openreview.net/forum?id=hl9ePdHO4_s (Last accessed on Sept. 20, 2024).

[Tat+19] Maxim Tatarchenko, Stephan R. Richter, Rene Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. “What Do Single-View 3D Reconstruction Networks Learn?” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA, June 2019, pp. 3400–3409. ISBN: 978-1-72813-293-8. DOI: 10/ghhg6c.

[Tha+13] Hendrik Thamer, Henning Kost, Daniel Weimer, and Bernd Scholz-Reiter. “A 3D-robot Vision System for Automatic Unloading of Containers”. In: *Proceedings of the IEEE Conference on Emerging Technologies Factory Automation (ETFA)*. Cagliari, Italy, Sept. 2013, pp. 1–7. DOI: 10.1109/ETFA.2013.6648028.

[Tha+14] Hendrik Thamer, Daniel Weimer, Henning Kost, and Bernd Scholz-Reiter. “3D-Computer Vision for Automation of Logistic Processes”. In: *Proceedings of the Efficiency and Innovation in Logistics*. Lecture Notes in Logistics. Cham, 2014, pp. 67–75. ISBN: 978-3-319-01378-7. DOI: 10.1007/978-3-319-01378-7_5.

[TMS11] Stephen Thomas, Bruce MacDonald, and Karl Stol. “Real-Time Robust Image Feature Description and Matching”. In: *Computer Vision – ACCV 2010*. Berlin, Heidelberg, 2011, pp. 334–345. ISBN: 978-3-642-19309-5. DOI: 10.1007/978-3-642-19309-5_26.

[TV19] Christopher S. Tang and Lucas P. Veelenturf. “The Strategic Role of Logistics in the Industry 4.0 Era”. In: *Transportation Research Part E: Logistics and Transportation Review* 129 (Sept. 2019), pp. 1–11. ISSN: 13665545. DOI: 10.1016/j.tre.2019.06.004.

[Uni19] Working Group United Nations. *Recommendations on the Transport of Dangerous Goods: Model Regulations*. 2019. URL: https://unece.

org/fileadmin/DAM/trans/danger/publi/unrec/rev21/ST-SG-AC10-1r21e_Vol1_WEB.pdf (Last accessed on Sept. 20, 2024).

[VCN15] Robert Varga, Arthur Costea, and Sergiu Nedevschi. "Improved Autonomous Load Handling with Stereo Cameras". In: *Proceedings of the IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*. Cluj-Napoca, Romania, Sept. 2015, pp. 251–256. ISBN: 978-1-4673-8200-7. DOI: 10.1109/ICCP.2015.7312639.

[VN14] Robert Varga and Sergiu Nedevschi. "Vision-Based Autonomous Load Handling for Automated Guided Vehicles". In: *Proceedings of the IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*. Cluj-Napoca, Romania, Sept. 2014, pp. 239–244. DOI: 10.1109/ICCP.2014.6937003.

[VN16] Robert Varga and Sergiu Nedevschi. "Robust Pallet Detection for Automated Logistics Operations:" in: *Proceedings of the Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. Rome, Italy, 2016, pp. 470–477. ISBN: 978-989-758-175-5. DOI: 10.5220/0005674704700477.

[VZ16] S Vargas-Osorio and C Zuniga. "A Literature Review on the Pallet Loading Problem". In: *Lámpsakos* 15 (2016). ISSN: 2145-4086. DOI: 10.21501/issn.2145-4086.

[Wah+19] Daniel Wahrmann, Arne-Christoph Hildebrandt, Christoph Schuetz, Robert Wittmann, and Daniel Rixen. "An Autonomous and Flexible Robotic Framework for Logistics Applications". In: *Journal of Intelligent & Robotic Systems* 93.3 (Mar. 2019), pp. 419–431. ISSN: 1573-0409. DOI: 10/gpc9mh.

[Wan+04] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. "Image Quality Assessment: From Error Visibility to Structural Similarity". In: *IEEE Transactions on Image Processing* 13.4 (Apr. 2004), pp. 600–612. ISSN: 1941-0042. DOI: 10.1109/TIP.2003.819861.

[Wan+18] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. “Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Vol. 11215. Cham, 2018, pp. 55–71. ISBN: 978-3-030-01251-9 978-3-030-01252-6. DOI: 10.1007/978-3-030-01252-6_4.

[Wan+22] Yonghui Wang, Wengang Zhou, Zhenbo Lu, and Houqiang Li. “UDoc-GAN: Unpaired Document Illumination Correction with Background Light Prior”. In: *Proceedings of the ACM International Conference on Multimedia*. MM ’22. New York, NY, USA, Oct. 2022, pp. 5074–5082. ISBN: 978-1-4503-9203-7. DOI: 10.1145/3503161.3547916.

[WCM22] Ratapol Wudhikarn, Phasit Charoenkwan, and Kanokwan Malang. “Deep Learning in Barcode Recognition: A Systematic Literature Review”. In: *IEEE Access* 10 (2022), pp. 8049–8072. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2022.3143033.

[Wei+10] F. Weichert, D. Fiedler, J. Hegenberg, H. Müller, C. Prasse, M. Roidl, and M. ten Hompel. “Marker-Based Tracking in Support of RFID Controlled Material Flow Systems”. In: *Logistics Research* 2.1 (June 2010), pp. 13–21. ISSN: 1865-0368. DOI: 10.1007/s12159-010-0025-6.

[Wen+19] Chao Wen, Yinda Zhang, Zhuwen Li, and Yanwei Fu. “Pixel2Mesh++: Multi-View 3D Mesh Generation via Deformation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea (South), Oct. 2019, pp. 1042–1051. ISBN: 978-1-72814-803-8. DOI: 10.1109/ICCV.2019.00113.

[WRV20] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. “Unsupervised Learning of Probably Symmetric Deformable 3D Objects From Images in the Wild”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA, 2020, pp. 1–10. DOI: 10.1109/CVPR42600.2020.00008.

[WRZ20] Manuel Woschank, Erwin Rauch, and Helmut Zsifkovits. "A Review of Further Directions for Artificial Intelligence, Machine Learning, and Deep Learning in Smart Logistics". In: *Sustainability* 12.9 (Jan. 2020), p. 3760. DOI: 10.3390/su12093760.

[WSB03] Z. Wang, E.P. Simoncelli, and A.C. Bovik. "Multiscale Structural Similarity for Image Quality Assessment". In: *Proceedings of the Asilomar Conference on Signals, Systems & Computers*. Vol. 2. Pacific Grove, CA, USA, Nov. 2003, 1398–1402 Vol.2. DOI: 10.1109/ACSSC.2003.1292216.

[Xia+13] Junhao Xiao, Jianhua Zhang, Benjamin Adler, Houxiang Zhang, and Jianwei Zhang. "Three-Dimensional Point Cloud Plane Segmentation in Both Structured and Unstructured Environments". In: *Robotics and Autonomous Systems* 61.12 (Dec. 2013), pp. 1641–1652. ISSN: 0921-8890. DOI: 10.1016/j.robot.2013.07.001.

[Xia+16] Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. "SUN Database: Exploring a Large Collection of Scene Categories". In: *International Journal of Computer Vision* 119.1 (Aug. 2016), pp. 3–22. ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-014-0748-y.

[Xia+17] Junhao Xiao, Huimin Lu, Lilian Zhang, and Jianhua Zhang. "Pallet Recognition and Localization Using an RGB-D Camera". In: *International Journal of Advanced Robotic Systems* 14.6 (Nov. 2017). ISSN: 1729-8814, 1729-8814. DOI: 10.1177/1729881417737799.

[Xie+19] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Shengping Zhang. "Pix2Vox: Context-aware 3D Reconstruction from Single and Multi-view Images". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea, July 2019. DOI: 10.1109/iccv.2019.00278.

[Xie+20] Guo-Wang Xie, Fei Yin, Xu-Yao Zhang, and Cheng-Lin Liu. "Dewarping Document Image by Displacement Flow Estimation with Fully Convolutional Network". In: *Proceedings of the Document*

Analysis Systems. Lecture Notes in Computer Science. Cham, 2020, pp. 131–144. ISBN: 978-3-030-57058-3. DOI: 10.1007/978-3-030-57058-3_10.

[Xie+21] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers". In: *Proceedings of the Conference on Neural Information Processing Systems*. Virtual, 2021. URL: https://openreview.net/forum?id=OG18MI5TRL (Last accessed on Sept. 20, 2024).

[XRT12] Jianxiong Xiao, Bryan Russell, and Antonio Torralba. "Localizing 3D Cuboids in Single-View Images". In: *Proceedings of the Advances in Neural Information Processing Systems*. Lake Tahoe, NV, USA, 2012, pp. 746–754. ISBN: 9781627480031. URL: https://dl.acm.org/doi/abs/10.5555/2999134.2999218 (Last accessed on Sept. 20, 2024).

[Xue+22] Chuhui Xue, Zichen Tian, Fangneng Zhan, Shijian Lu, and Song Bai. "Fourier Document Restoration for Robust Document Dewarping and Recognition". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA, June 2022, pp. 4563–4572. ISBN: 978-1-66546-946-3. DOI: 10.1109/CVPR52688.2022.00453.

[Yan+19] Bo Yang, Jianan Wang, Ronald Clark, Qingyong Hu, Sen Wang, Andrew Markham, and Niki Trigoni. "Learning Object Bounding Boxes for 3D Instance Segmentation on Point Clouds". In: *Proceedings of Advances in Neural Information Processing Systems*. Vol. 32. Vancouver, Canada, 2019. URL: https://dl.acm.org/doi/10.5555/3454287.3454892 (Last accessed on Sept. 20, 2024).

[Yan+21] Shuo Yang, Min Xu, Haozhe Xie, Stuart Perry, and Jiahao Xia. "Single-View 3D Object Reconstruction from Shape Priors in Memory". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA, June

2021, pp. 3151–3160. ISBN: 978-1-66544-509-2. DOI: 10.1109/cvpr46437.2021.00317.

[Ye+22] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven C. H. Hoi. "Deep Learning for Person Re-Identification: A Survey and Outlook". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.6 (June 2022), pp. 2872–2893. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3054775.

[YS19] S. Yang and S. Scherer. "CubeSLAM: Monocular 3-D Object SLAM". In: *IEEE Transactions on Robotics* 35.4 (Aug. 2019), pp. 925–938. ISSN: 1941-0468. DOI: 10/gg652k.

[Yu+18] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. "Deep Layer Aggregation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA, June 2018, pp. 2403–2412. ISBN: 978-1-5386-6420-9. DOI: 10.1109/CVPR.2018.00255.

[Zak+21] Sergey Zakharov, Rares Andrei Ambrus, Vitor Campagnolo Guizilini, Dennis Park, Wadim Kehl, Fredo Durand, Joshua B. Tenenbaum, Vincent Sitzmann, Jiajun Wu, and Adrien Gaidon. "Single-Shot Scene Reconstruction". In: *Proceedings of the Annual Conference on Robot Learning*. Proceedings of Machine Learning Research. London, UK, Nov. 2021, pp. 501–512. URL: https://proceedings.mlr.press/v164/zakharov22a.html (Last accessed on Sept. 20, 2024).

[ZC88] Zhou and Chellappa. "Computation of Optical Flow Using a Neural Network". In: *Proceedings of the IEEE International Conference on Neural Networks*. Vol. 2. San Diego, CA, USA, July 1988, pp. 71–78. DOI: 10.1109/ICNN.1988.23914.

[Zen+17] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Ed Walker Jr, Alberto Rodriguez, and Jianxiong Xiao. "Multi-View Self-Supervised Deep Learning for 6D Pose Estimation in the Amazon

Picking Challenge". In: *Proceedings of the IEEE International Conference on Robotics and Automation*. Singapore, May 2017. DOI: 10.1109/ICRA.2017.7989165.

[Zha+18] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Salt Lake City, Utah, USA, 2018, pp. 586–595. DOI: 10.1109/cvpr.2018.00068.

[Zha+19] Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu, and Xindong Wu. "Object Detection With Deep Learning: A Review". In: *IEEE Transactions on Neural Networks and Learning Systems* 30.11 (Nov. 2019), pp. 3212–3232. ISSN: 2162-2388. DOI: 10.1109/TNNLS.2018.2876865.

[Zhe+20] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. "Structured3D: A Large Photo-realistic Dataset for Structured 3D Modeling". In: *Computer Vision – ECCV 2020*. Lecture Notes in Computer Science. Cham, 2020, pp. 519–535. ISBN: 978-3-030-58610-2. DOI: 10.1007/978-3-030-58545-7_30.

[Zho+21] Benchun Zhou, Aibo Wang, Jan-Felix Klein, and Furmans Kai. "Object Detection and Mapping with Bounding Box Constraints". In: *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. Karlsruhe, Germany, Sept. 2021, pp. 1–6. DOI: 10.1109/MFI52462.2021.9591174.

[Zou+23] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. "Object Detection in 20 Years: A Survey". In: *Proceedings of the IEEE* (2023), pp. 1–20. ISSN: 1558-2256. DOI: 10.1109/JPROC.2023.3238524.

Publications by the Author

[Dör+20a] Laura Dörr, Felix Brandt, Martin Pouls, and Alexander Naumann. "An Image Processing Pipeline for Automated Packaging Structure Recognition". In: *Forum Bildverarbeitung*. Karlsruhe, Germany, 2020, pp. 239–251. ISBN: 978-3-7315-1053-6. DOI: 10.5445/KSP/1000124383.

[Dör+20b] Laura Dörr, Felix Brandt, Martin Pouls, and Alexander Naumann. "Fully-Automated Packaging Structure Recognition in Logistics Environments". In: *Proceedings of the International Conference on Emerging Technologies and Factory Automation*. Vienna, Austria, Sept. 2020. ISBN: 978-1-72818-956-7. DOI: 10.1109/ETFA46521.2020.9212152.

[Dör+21] Laura Dörr, Felix Brandt, Alexander Naumann, and Martin Pouls. "TetraPackNet: Four-Corner-Based Object Detection in Logistics Use-Cases". In: *Proceedings of the DAGM German Conference on Pattern Recognition*. Bonn, Germany, 2021. ISBN: 978-3-030-92659-5. DOI: 10.1007/978-3-030-92659-5_35.

[Dör+23] Laura Dörr, Katharina Glock, Felix Brandt, Alexander Naumann, and Martin Pouls. "A Digital Measuring and Load Planning System for Large Transport Assets". In: *2023 International Scientific Symposium on Logistics: Conference Volume*. June 2023, pp. 49–55. DOI: 10.25366/2023.124.

[Gri+23] Daniel Grimm, Maximilian Zipfl, Felix Hertlein, Alexander Naumann, Jürgen Lüttin, Steffen Thoma, Stefan Schmid, Lavdim Halilaj,

Achim Rettinger, and J. Marius Zöllner. “Heterogeneous Graph-based Trajectory Prediction Using Local Map Context and Social Interactions”. In: *Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC)*. Bilbao, Spain, Sept. 2023.

[HN23] Felix Hertlein and Alexander Naumann. “Template-Guided Illumination Correction for Document Images with Imperfect Geometric Reconstruction”. In: *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. Paris, France, 2023, pp. 904–913. DOI: 10.1109/ICCVW60793.2023.00097.

[HNP23] Felix Hertlein, Alexander Naumann, and Patrick Philipp. “Inv3D: A High-Resolution 3D Invoice Dataset for Template-Guided Single-Image Document Unwarping”. In: *International Journal on Document Analysis and Recognition (IJDAR)* (Apr. 2023). ISSN: 1433-2825. DOI: 10.1007/s10032-023-00434-x.

[Nau+20] Alexander Naumann, Laura Dörr, Niels Ole Salscheider, and Kai Furmans. “Refined Plane Segmentation for Cuboid-Shaped Objects by Leveraging Edge Detection”. In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Miami, Florida, USA, Dec. 2020, pp. 432–437. DOI: 10.1109/ICMLA51294.2020.00096.

[Nau+22] Alexander Naumann, Felix Hertlein, Benchun Zhou, Laura Dorr, and Kai Furmans. “Scrape, Cut, Paste and Learn: Automated Dataset Generation Applied to Parcel Logistics”. In: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*. Nassau, Bahamas, Dec. 2022, pp. 1026–1031. DOI: 10.1109/ICMLA55696.2022.00171.

[Nau+23a] Alexander Naumann, Felix Hertlein, Laura Dörr, and Kai Furmans. “Parcel3D: Shape Reconstruction from Single RGB Images for Applications in Transportation Logistics”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*

Workshops (CVPRW). Vancouver, Canada, June 2023, pp. 4402–4412. DOI: 10.1109/cvprw59228.2023.00463.

[Nau+23b] Alexander Naumann, Felix Hertlein, Laura Dörr, Steffen Thoma, and Kai Furmans. *Literature Review: Computer Vision Applications in Transportation Logistics and Warehousing*. Preprint. Apr. 2023. arXiv: 2304.06009. URL: https://arxiv.org/abs/2304.06009 (Last accessed on Sept. 20, 2024).

[Nau+23c] Alexander Naumann, Felix Hertlein, Daniel Grimm, Maximilian Zipfl, Steffen Thoma, Achim Rettinger, Lavdim Halilaj, Juergen Luettin, Stefan Schmid, and Holger Caesar. "Lanelet2 for nuScenes: Enabling Spatial Semantic Relationships and Diverse Map-Based Anchor Paths". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Vancouver, BC, Canada, June 2023, pp. 3247–3256. DOI: 10.1109/CVPRW59228.2023.00327.

[Nau+24] Alexander Naumann, Felix Hertlein, Laura Dörr, and Kai Furmans. "TAMPAR: Visual Tampering Detection for Parcel Logistics in Postal Supply Chains". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Waikoloa, Hawaii, USA, Jan. 2024, pp. 8076–8086.

[NKS18] Alexander Naumann, Oliver Kolb, and Matteo Semplice. "On a Third Order CWENO Boundary Treatment with Application to Networks of Hyperbolic Conservation Laws". In: *Applied Mathematics and Computation* 325 (May 2018), pp. 252–270. ISSN: 0096-3003. DOI: 10/ggqbkm.

Supervised Theses

Zeyu Wang, *"Image-based 3D Shape Reconstructions of Logistics Objects"*, Master's Thesis, Karlsruhe Institute of Technology, Institute for Material Handling and Logistics, Karlsruhe, Germany, Nov. 2020. Supervised jointly with Laura Dörr.

Adrian Schneider, *"Verfeinerung der Segmentierung von Paket-Seitenflächen aus Bilddaten"*, Bachelor's Thesis, Karlsruhe Institute of Technology, Institute for Operations Research, Karlsruhe, Germany, Jan. 2022.

Jonathan Sexton, *"Evaluation of Neural Radiance Fields (NeRF) as a Method for 3D Reconstruction in the Field of Logistics"*, Master's Thesis, University of Bath, Department of Computer Science, Bath, United Kingdom, Apr. 2023. Supervised externally.

Sonya Voneva, *"Improving Table Structure Recognition for Geometrically Distorted Tables using Table Segmentation and Rectification"*, Master's Thesis, Karlsruhe Institute of Technology, Institute of Information Security and Dependability, Karlsruhe, Germany, Jan. 2024. Supervised jointly with Felix Hertlein.