robotics

MDPI

Article

A 3D World Interpreter System for Safe Autonomous Crane Operation

Frank Bart ter Haar *,†, Frank Ruis †and Bastian Thomas van Manen †

TNO—Intelligent Imaging, Oude Waalsdorperweg 63, 2597 AK The Hague, The Netherlands; frank.ruis@tno.nl (F.R.); bastian.vanmanen@tno.nl (B.T.v.M.)

* Correspondence: frank.terhaar@tno.nl

† These authors contributed equally to this work.

Abstract: In an effort to improve short-sea shipping in Europe, we present a 3D world interpreter (3DWI) system as part of a robotic container-handling system. The 3DWI is an advanced sensor suite combined with AI-based software and the communication infrastructure to connect to both the crane control and the shore control center. On input of LiDAR data and stereo captures, the 3DWI builds a world model of the operating environment and detects containers. The 3DWI and crane control are the core of an autonomously operating crane that monitors the environment and may trigger an emergency stop while alerting the remote operator of the danger. During container handling, the 3DWI scans for human activity and continuously updates a 3D-Twin model for the operator, enabling situational awareness. The presented methodology includes the sensor suite design, creation of the world model and the 3D-Twin, innovations in AI-detection software, and interaction with the crane and operator. Supporting experiments quantify the performance of the 3DWI, its AI detectors, and safety measures; the detectors reach the top of VisDrone's leaderboard and the pilot tests show the safe autonomous operation of the crane.

Keywords: 3D scene reconstruction; digital twinning; object detection; deep learning; autonomous systems; short-sea shipping

Citation: ter Haar, F.B.; Ruis, F.; van Manen, B.T. A 3D World Interpreter System for Safe Autonomous Crane Operation. *Robotics* **2024**, *13*, 23. https://doi.org/10.3390/robotics13020023

Academic Editor: Chris Lytridis

Received: 20 December 2023
Revised: 20 January 2024
Accepted: 21 January 2024
Published: 26 January 2024

1. Introduction

This paper presents a 3D world interpreter (3DWI) system as part of a robotic container-handling system (RCHS) to automatically locate containers to pick up whilst safeguarding human activity in the scene. The research carried out in this paper has a strong focus on the robust detection and classification of human activity in future harbor scenarios. The RCHS with 3DWI in combination with an intelligent operator support system (IOSS) for shore control centers is one of the innovations in the MOSES EU-project to improve short-sea shipping (SSS) in Europe (https://moses-h2020.eu/, accessed on 20 January 2024). The overall program contributes to the rapidly growing application of artificial intelligence (AI) and autonomous systems in the maritime industry. In this logistic use case, the autonomous vessels transfer containers between international harbors and harbors without a container-handling infrastructure with an on-board RCHS for autonomous on- and off-loading. The 3DWI concept contributes to the safe autonomous operation of the RCHS. The current design of the 3DWI assumes a rigidly placed crane, and the conducted tests are with the RCHS on a test terrain.

The need for innovations in this context is explained by Ghaderi [1] and includes trends, feasibility, and implications of autonomous technologies in SSS. It shows that these technologies are viable to the challenges that the shipping industry is facing in terms of crew cost and skill shortage, and that a network of unmanned vessels (with an on-board RCHS) operated from a shore control center is a good approach. The point of attention is the safety, operability, and liability in smaller-scale SSS harbor scenarios.

In our system of systems, the operators of a shore control center use the IOSS to supervise a network of vessels equipped with autonomous RCHS (Figure 1). The RCHS has a crane control unit (CCU) for the mechanical pick and place of containers and a 3DWI to (1) 3D scan the environment to guide the CCU to the containers and (2) to detect obstacles and human activity, (in that case) stop the crane, update the 3D twin of the IOSS, and ask the operator for assistance. One of the functions of IOSS is to support an operator with a 3D-Twin for situation awareness and remote tools to solve occurring issues of the autonomous crane, e.g., due to sensor failures, wrong container configuration, or detected obstacles. The system of systems is used in pilot tests, and results are reported in this work. In these tests, the crane, CCU, and 3DWI systems were located on a test terrain in Örnsköldsvik, Sweden, and the 3D-Twin, IOSS, and operators in the immersive collaboration lab in Soesterberg, The Netherlands, with merely a 4G connection and a communication server in between.

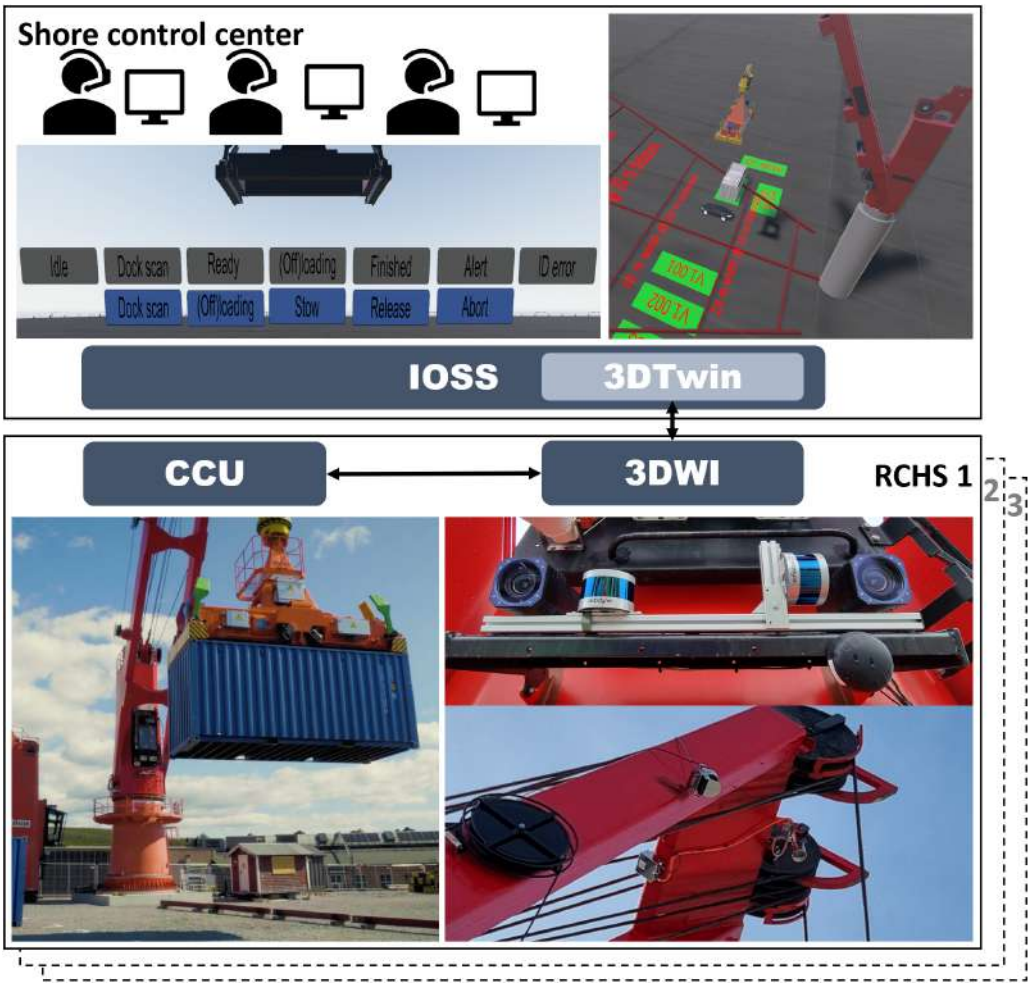

Figure 1. System of systems, including the crane control unit (CCU) and 3D world interpreter (3DWI) with sensor suite as part of a RCHS and the 3D-Twin visualization of this RCHS as part of IOSS in the shore control center. Multiple robotic cranes can be remotely monitored and assisted by the operator.

In this paper, the focus is on the design of the 3DWI system, its 3D-Twin construction, and the use of AI object detectors to safeguard both the RCHS and human activity in the SSS harbor scenarios. For the 3DWI, an advanced sensor suite is selected based on light detection and ranging (LiDAR), stereo vision, and an overhead camera as the basis for the detectors and to update the 3D-Twin for the remote operators. After the related work in Section 2, the methodology of the 3DWI is described in Section 3, followed by details on safety-critical components of the 3DWI in Section 4, experiments and results in Section 5, discussions in Section 6 and conclusions in Section 7.

2. Related Work

Related to the safety concept of our autonomous crane, Tiusanen et al. [2], in their work, elaborate on three approaches to standardized safety: (1) The strict separation of the autonomous system and the controlled access to the work area; (2) A non-separated working areas from humans, machines, and autonomous machines where the autonomous system carry a sensor system for the safety; (3) The reliance on a local or remote operator to stop the autonomous system when a risk occurs. In our work, a mixture of these approaches is used. The RCHS is by design mounted on a vessel without other moving objects, but interacts in a non-separated working area (the docks). To this end, the RCHS carries a sensor system with AI technology to stop its process when a problematic situation occurs, and in those cases, the remote operator is responsible for a safe solution and continuation of the autonomous process.

The safety of autonomous systems, and vehicles in particular, is an extensively researched area [3,4], and the role of AI and machine learning techniques in these systems is under debate [5,6]. The use of a digital twin of a data-driven autonomous vehicle in its environment [7] or a digital twin of its decision process [8] helps to understand the actions of the AI and helps improve the system's safety and security. Similar to Almeaibed et al. [7], our digital twin encompasses the 3D visual representation of physical entities, such as the crane, containers, and environment, and we refer to it as 3D-Twin. Benchmarks and surveys are another way to objectively measure the status of data-driven AI for vehicle autonomy [9,10]. Similar to the KITTI-360 benchmark [9], our sensor suite comprises a stereo camera, LiDARs, and an overhead camera combined with AI algorithms for 3D analysis. However, the methods in these benchmarks are tuned at eye-level sensor capture and do not apply to our domain, where the sensor suite looks down from 10 to 30 m in height and senses up to 50 m in distance. Studies [11,12] have shown that certain weather conditions such as rain, fog, or snow decrease the performance of the LiDAR sensors, especially in the case of severe fog and heavy rain [13,14]. LiDAR sensors alone would not be able to guarantee safety in all types of weather conditions. Poor environmental conditions also affect the quality of the camera images, making object detection harder. However, research in the field of autonomous driving has shown that additional training data and improvements in the model architecture allow the model to better generalize and significantly improve object detection accuracy in these conditions [15,16].

Object detection from a conventional viewpoint is presently dominated by models with large amounts of parameters [17–21], often incorporating transformer backbones or prediction heads [22]. Detecting objects such as people and vehicles from an overhead view at high altitudes, as is the case for our crane-mounted camera, is a challenging problem where objects could be as small as 10 pixels in area, with limited public availability of high-quality image datasets. The best-performing models for conventional viewpoints do not perform well in this small object detection domain [23] due to factors like the computational costs of increasing the input resolution, as well as being over-parameterized for small objects that lack semantic features. Most existing methods tackle this problem through specialized architectures [24–26], though these are often still computationally expensive and not as thoroughly tested and understood as more established methods, such as the you only look once (YOLO) series [27–30]. Closely related to our use case, Golcarenarenji et al. [31] detail a method for safety monitoring on complex industrial sites with a crane-mounted camera. Their method relies on a custom architecture optimized for real-time detection of small objects trained with a large private dataset. Other detection methods using crane-mounted cameras, attempt to tackle the data scarcity issue through the use of synthetic data [32–34], which introduces a significant unsolved challenge in overcoming the domain gap between realistic imagery, as well as a large initial development effort to recreate the target environments.

Considering the deep learning methods in particular, He et al. [35] show with bag of tricks that simply aggregating various methods and applying them to a ResNet-50 can increase its classification performance to be on par or even better than much more recent

architectures. Similar results are achieved for other architectures as well [36,37]. We do the same for a YOLOv5 object detector, details of which can be found in Section 4.2. Our proposed approach for human activity detection has several benefits over the specialized methods in the existing literature: (1) An established architecture is used that has a proven track record, and its limitations and strengths are well-understood; (2) Many of the aforementioned tricks already exist as options in the code base or can be implemented in a single line of code; (3) The YOLOv5 code base is updated frequently, with new releases being rigorously tested by the open-source community. Research-only publications such as YOLOv7 [30], in contrast, are usually not updated after their initial publication date; (4) The increased robustness brought about by our pipeline allows the use of publicly available training data with less similarity to our target domain.

More recently, with the advent of autonomous driving, 3D object detection algorithms have become faster and more accurate but still require sufficiently dense point clouds, especially for classification tasks [38,39]. The point cloud that typically results from long-range LiDARs is too sparse for accurate 3D detection and classification in static conditions. The fusion of LiDAR and stereo point clouds can produce sufficiently dense point clouds. However, our previous attempts to combine stereo algorithms with 3D object detection proved to be slower and more complex [40] compared to our current approach that combines YOLO applied to 2D images with the 3D LiDAR point cloud.

3. 3DWI Methodology

This section describes the design and methodology of the 3DWI system and the 3D-Twin systems and their components and interfaces. The 3DWI is a system that (1) enables the RCHS to scan and interpret the harbor environment for safe autonomous operations and (2) gathers information about the environment in the 3D-Twin for the remote operator. The 3D-Twin assists the situation awareness build-up for the remote operator, who monitors the autonomous operation, solves occurring issues, and verifies the completion of tasks. The 3DWI comprises a sensor suite for long-distance sensing, the algorithms to scan the environment in 3D and to extract 3D containers and obstacles, the algorithms to detect human activity in 3D, and last but not least, the 4G network connectivity to continuously share the sensed results with the 3D-Twin for the operators and IOSS. The visual overview of the methodology is shown in Figure 2, and the processing steps are described in this section.

3.1. Sensor Suite

The sensor suite designed for this work is mounted on the crane base at a 45° downwards angle. It includes a fixed rig with two 5 MP cameras (2560 × 2048 resolution) in a stereo setup and 8 mm lenses to capture with 74° horizontal field of view (FOV) and 62° vertical FOV, a sufficiently wide area for detecting objects. For high-accurate 3D measurements at fifty meter distance, two VLP-16s are added in both a horizontal and vertical setup. The sensor suite is calibrated such that 3D LiDAR points are projected to the 2D stereo camera images to retrieve the red, green, blue (RGB) color for the LiDAR points. To mimic the real crane, the rig with these sensors is mounted on a pan-tilt unit (PTU) that is controlled by the 3DWI during the scan procedure.

In the foreseen design of a vessel-mounted crane, the sensor suite is at 13 m height on the rotating part of the crane with a pitch of 45° downward. In this configuration, the camera captures the ground in front of the crane, starting from 3.2 m up to 50 m with, respectively, a width of 17.5 up to 67 m on the ground. In the pilot tests, the sensor suite also has a gravity-aligned zoom camera (but typically set to 40° hFOV) near the crane's jib top to have an overhead view on and around the containers for the 2D/3D object detection module. For each 3DWI sensor, the placement (in mm), hFOV, resolution, and captured area are shown in Figure 3; from this, the pixels on target can be derived, e.g., 106 pixels in the stereo camera for a 1 m object at 10 m from the crane.

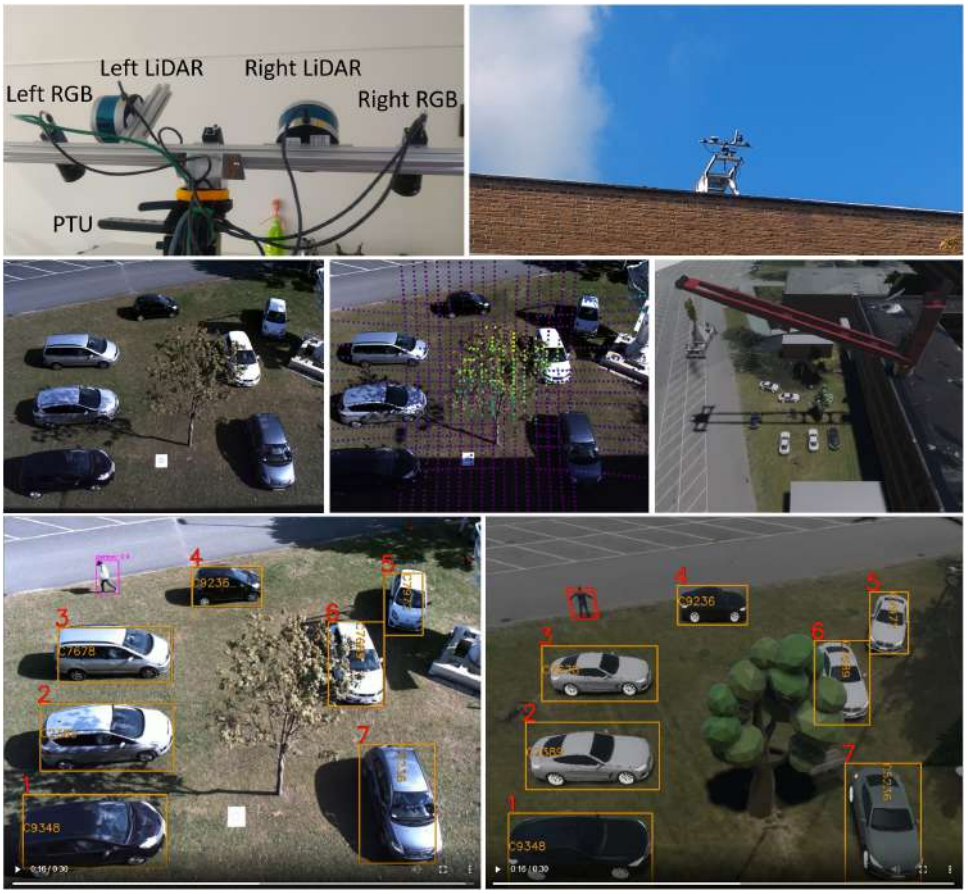

Figure 2. The first prototype of the 3DWI system for the rooftop recordings: The sensor suite on a pan-tilt unit (**top row**). The left camera view, projected LiDAR points, and the 3D-Twin of the rooftop environment and crane (**middle row**). The human activity detection and visualization in the 3D-Twin for the operator with corresponding object IDs (**bottom row**).

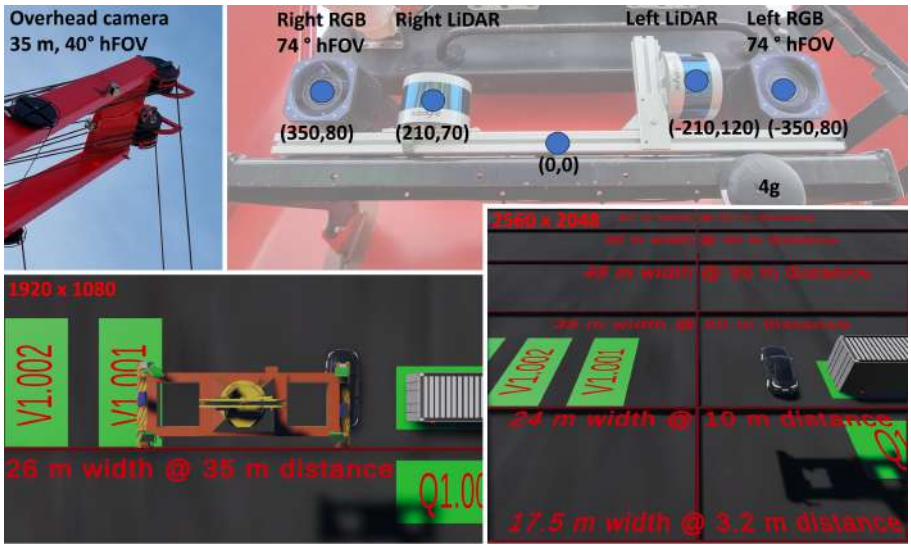

Figure 3. The configuration of the 3DWI sensors and the area captured by the overhead and stereo cameras.

In practice, during the rooftop recordings, the sensor suite was at 11 m height, and during the pilot tests, the sensor suite was at only 6 m height due to a small pedestal. To

compensate for the lower altitude in the pilot tests, the sensor suite was set to a pitch of 28° downward; not the ideal coverage, but sufficient for the tests.

3.2. 3D Obstacle Map

When the (future) autonomous vessel with the RCHS arrives at the harbor, the 3DWI scans the dock with its sensor suite and slowly increases the heading of the crane to capture the full range. The highly detailed (and colored) 3D LiDAR measurements are concatenated and merged into a ground-plane aligned point cloud and then mapped into a digital elevation map (DEM). The DEM is 800 × 400 pixels large to store the 3D measurements of 160 × 80 meters of the test terrain with cells of 2 × 2 decimeters and keeping the highest value. From this DEM, the 3D obstacles are segmented as either 20 ft containers (based on 3D shape) or no-go areas that the crane should avoid during (off-)loading. These no-go areas per harbor may also be indicated by the operator beforehand. For each obstacle and container extracted from the DEM, the bounding box center is used to select the closest 3D coordinate from the concatenated LiDAR measurements. Figure 4 illustrates a 3D scan captured by the 3DWI system at the test site and the resulting container segmentation masks.

Next, the 3D position and orientation of the containers and obstacles are sent to the 3D-Twin for the operator to check, e.g., does the situation match the work order? Is the number of containers correct? Are there more no-go areas to add? Finally, the 3DWI and the CCU start the (off-)loading procedure while continuously updating the container configuration (position and orientation) in the 3DWI and 3D-Twin based on the live state information of the crane. In this mode, the 3DWI starts the safeguarding of the vicinity with live 2D/3D object detection.

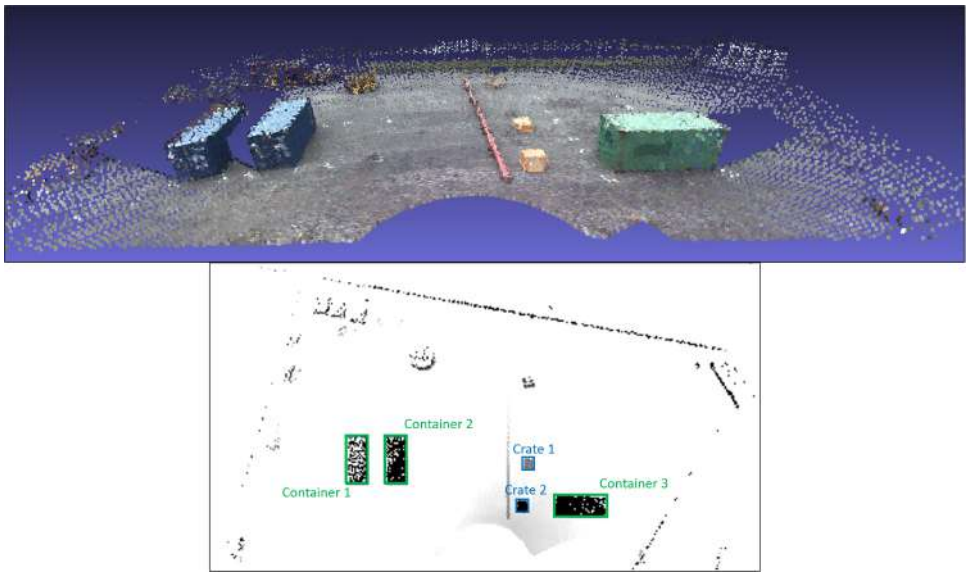

Figure 4. Colored 3D LiDAR measurements captured by the 3DWI system at a test site with three containers (**top**) and the constructed DEM (**bottom**) with the segmented containers (green) and wooden crates (blue).

3.3. 2D/3D Object Detection

The live object detection module uses images from the stereo camera and overhead camera. To this footage, two detection models are applied: a generic object detector YOLOv5 [29] and an optimized human activity detector for both oblique and overhead view. The human activity detector is a custom YOLOv5 model specifically trained to have very high accuracy on critical classes such as persons and vehicles (detailed explanation in Section 4). On the other hand, the generic object detector is trained on the COCO dataset; hence, it can detect up to 80 different classes, but with a lower accuracy. Since not all COCO

classes are relevant, we have condensed them into six categories: small, medium, and large objects; animals; vehicles; and persons. The benefit of the generic object detector lies in the diversity of classes. By combining the detections of both models, the 3DWI is able to detect a large range of classes with very high accuracy for persons and vehicles. Next, the 2D detections are converted to 3D detection using the one-to-one mapping of the calibrated sensor suite from the 2D camera to the 3D LiDAR data. 3D points that coincide with the 2D detection box are averaged and provide a good estimate of the 3D target location of both generic and human-specific objects. It may happen that a container is also detected by the generic object detector or that the generic detector detects objects in the no-go area or that a person is detected in a no-go (safe-zone) area. These cases are solved in the final reasoning step for both the oblique and overhead camera analysis. The flow of the detections based on the (static) 3D obstacle map and the live 2D/3D object detection is shown in Figure 5. The final set of 3D objects and red alerts (i.e., the human activity detection) are continuously sent to the CCU and 3D-Twin.

Our state-of-the-art contribution to human activity detection is described in Section 4. Human activity detection has a central place in this paper because of its safeguarding function. Upon detection within a fifty-meter distance of the crane, the 3DWI raises a red alert that logically forces the autonomous crane (via the CCU) to an emergency stop and requests IOSS to intervene. Therefore, our system of systems requires a high detection rate and low miss rate for this detector in particular.

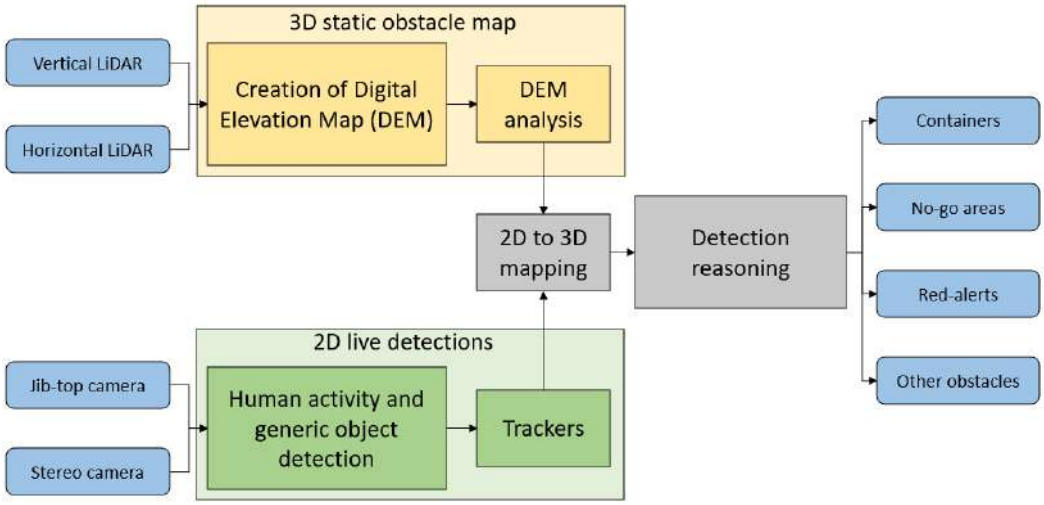

Figure 5. Flow chart of 3DWI to compute a 3D obstacle map and do 2D/3D objects detection. The 3D position and orientation of containers, generic obstacles, red alerts, and no-go obstacles are shared with CCU and 3D-Twin.

3.4. 3D Twin

The 3D-Twin is the main interface between the operator and the autonomous crane. It renders the (pre-)captured remote environment, the live state of the crane, the recent dock scan, the detected objects (containers, no-gos, obstacles, red alerts), the current state of the autonomous crane operation, and basic controls. Collectively, this presents a detailed situational awareness for the operator tasked with overseeing the 3DWI and CCU systems. The 3D-Twin for the rooftop recordings is shown in Figure 2. A more complete 3D-Twin, shown in Figure 6, was created for the pilot tests with the RCHS. In this situation, the crane is slewing from the left to the right side when suddenly a person is detected (person with red bounding box to the right of the spreader), and the system comes to a halt. Several features improving the operator situational awareness can be observed in Figure 6. First, the 3D-Twin visualizes the states of the real crane in its lateral and vertical rotations, as well as those of the hoisting mechanism and the spreader. Second, all detected 3D objects by the 3DWI are rendered in real-time in the 3D-Twin, including containers and potential

threats such as persons. Third, objects that are classified as human activity (i.e., persons and vehicles) are visualized with a 3D red bounding box to highlight their presence in the 3D-Twin to the operator. Subsequently, the system jumps to an alert state visible by the system state panels at the top in Figure 6. The operator is then asked to resolve the situation. Additionally, the operator can request and inspect the most recent oblique and overhead image from the 3DWI sensor suite to verify the presence of the threat. Fourth, on the left-hand side of the screen in Figure 6, the operator has access to the current crane work order list. In this manner, the operator can verify the crane's operation. A work order, which is a container displacement from one container slot to another, is crossed off from the list (by the 3DWI) when the movement is completed. The weight of the live container is also displayed under the current work order for the operator to see. Fifth, the system receives from the CCU the container ID read as the container is about to be picked up. The 3D-Twin renders this container ID on the container sides, top, and bottom. Conversely, if the CCU detects a mismatch in the expected container ID, an error flag is raised, which will be communicated to the operator again through the 3D-Twin system state panels. The responsible operator can release an error when deemed safe using the blue buttons under the state panels in Figure 6.

Figure 6. Snapshot of the 3D-Twin for an operational environment with actual containers, showing the spreader with its container, a detected person, the camera's FOV in dark gray, the current work order on the light gray panel, and the system states at the top of the image and blue buttons below for the human interaction.

4. 3DWI Safety-Critical Experiments

This section describes the conducted experiments to evaluate the safety of the 3DWI system in the context of autonomous crane operation in open harbors. Essential for safety monitoring are the human activity detector, the detection tracking, the end-to-end processing speed, and the crane's emergency stop. The key element for the safe operation is the human activity detector applied to the camera images. The following subsections elaborate on the choices and improvements related to the detection architecture, the training dataset, the type of model, and additional parameter options. In Section 5, the experiments conducted and their results are explained.

4.1. Datasets

Three classes of datasets are used for this paper. The first is a series of oblique rooftop recordings (Figure 2) with the stereo and LiDARs on a pan-unit to test and develop the 3DWI and 3D-Twin systems. The second is a series of public datasets, specifically

VisDrone [41], HERIDAL [42], and MOCS [43], to create a robust human activity detector. The third is a set of pilot demonstration recordings with the stereo and LiDARs mounted on the crane base and an overhead camera on the crane's jib top.

4.2. Baseline Detection Architecture

For our baseline deep learning detector, we start with the code base from YOLOv5 (https://github.com/ultralytics/yolov5, accessed on 20 January 2024), an object detection framework based on the YOLO [27,28,44] series. YOLOv5 is a robust baseline where many elements of the bag of tricks paper [35] have already been implemented and extensively evaluated on the COCO object detection benchmark [45]. For instance, mixed precision training [46], Mixup [47], and CSPNet [48] are available. Unless otherwise stated, the default training configuration is used in this work. As the library is updated almost daily, we link to the state of the repository at the specific point in time it was pulled from GitHub (https://github.com/ultralytics/yolov5/tree/c9042dc2adbb635aeca407c10cf492a6eb14d772 , accessed on Januari 2024). This baseline model is referenced in Section 5.1 and 5.2. The baseline COCO model is used in Section 5.2, to show the importance of a dedicated human activity detector.

All custom models are trained with the default train script provided in the YOLOv5 repository. The only change in the code that needs to be added is BlurPool, as described in the anti-aliasing section of Section 4.5.

4.3. Dataset Tuning to Application

Using FiftyOne [49], a computer vision dataset visualization and management tool, several widely used datasets were inspected for their applicability to small object detection in overhead camera views considering the FOV of the overhead camera in the jib top. The VisDrone [41] dataset was selected as the highest quality dataset closest to our target domain. For this dataset, it is important to use the ignore regions provided in the original label files to avoid typical label noise and degraded performance. Note that the YOLOv5 code base silently discards the ignore regions in its VisDrone example code.

The original VisDrone labels contain fine-grained categories such as "pedestrians", which contains humans standing upright, and "people", which contains humans in sitting or crouching positions, typically when driving a bike or motorcycle. Other fine-grained classes are "car" and "van", and "tricycle" and "awning tricycle". For our method, these distinctions are too specific. By merging these classes into more generic person and vehicle classes and by removing the tricycle class, our model does not need to learn these distinctions and is no longer affected by fine-grained label noise.

Finally, while the VisDrone dataset is captured in a wide variety of locations and viewpoints, it is still limited to mainly urban areas in Chinese cities. According to our preliminary findings, this affects the model's generalization capability, resulting in many false positives with high confidence in the presence of high contrast shadows in the snow or sunlight reflected in the water, as well as false negatives in, e.g., a harbor environment where people may wear high visibility clothing and helmets (shown in Figure 7). To have the model generalize to shadows, snow, and sunlight, the training data are extended with the HERIDAL [42] dataset, which contains a small set of high-resolution drone images with persons in wilderness scenes such as forested mountains or snowy plains. To generalize to harbor environments, the dataset is extended with a subset of the MOCS [43] dataset, an object detection dataset of remote construction site locations. Not all 41k MOCS images are added; to acquire mostly drone-captured images from this set, those with more than eight labeled objects are selected.

The benefits of the improved dataset and training procedure are shown in Figure 7. It shows progressively more accurate detection results for oblique (top row) and overhead view (bottom row) images that are out of the training distribution. These views are representative of the pilot demonstration, but too far out of the domain for a standard YOLOv5 detector pre-trained on COCO (first detection column). The same detector trained

on VisDrone data (middle detection column) performs better at oblique angles but with misses in both oblique and overhead views. Our training procedure for the same model and dataset (right detection column) significantly increases performance from both points of view.

The tuned dataset described in this section is referred to in Section 5.1 and is important to make the trained model applicable to our captured rooftop and pilot demonstration datasets.

Figure 7. Progressively more accurate detection results for oblique (**top row**) and overhead view (**bottom row**) images that are out of the training distribution. More red bounding boxes means that more persons are correctly detected. From left to right, the results of COCO pretrained, VisDrone trained, and our trained detector.

4.4. Alternative Detection Architectures

Recent state-of-the-art results on VisDrone [24] opt for swapping out parts of existing architectures with more complex transformer-based blocks or more complex necks such as multiple BiFPN blocks [50]. As most methods do not publish their code, we focus our comparisons on TPH-YOLOv5 [25], a method that came within 0.25% of the 2021 VisDrone challenge first-place winner. After the challenge, the authors updated their code with a better-performing architecture. This version is used in our experiments. Similarly, YOLOv7-Drone [51] is an adaptation of YOLOv7 geared towards improving the detection of small objects. Since their codebase is not yet public, we only provide the results reported in their paper in our comparison.

Another recent method is PP-YOLOE-SOD, which is an official extension of the recent PP-YOLOE [26] specifically tuned for small object detection, including an anchor-free detector head, distributed focal loss, and the replacement of convolution blocks in the neck with transformer blocks. In the work of Northcutt et al. [52], the observation is made that less complex models can outperform larger models when the label noise in the test set is reduced. Similarly, we hypothesize that these more complex models are over-fitted to the (VisDrone) data, and simpler models generalize better in the case of domain shifts. These alternative TPH-YOLOv5, YOLOv7-Drone, and PP-YOLOE-SOD architecture models are compared to the baseline model and tuned baseline models in Section 5.1. For comparison, our results on the original VisDrone data are reported, as it is the current state-of-the-art on VisDrone.

4.5. Tuned Baseline Detection Architecture

As mentioned, YOLOv5 is a robust baseline architecture with many improvements from the literature implemented and validated [35], but often not enabled by default. The following paragraphs elaborate on the conducted changes to create our tuned baseline architecture, which together with the tuned dataset for training, forms our final human activity detector.

Extra prediction head: YOLOv5 comes with an optional configuration for an extra prediction head at a 4×4 pixel resolution, also called the P2 head (found in the yolov5-p2.yaml file). Although simply up-scaling the input image resolution usually performs better in terms of performance gain versus computation cost, we find that a P2 head is worth including if there still is a computational budget left once the input resolution has been saturated.

Exponential moving average: Also implemented in YOLOv5 by default is model EMA, which keeps an exponential moving average of the model weights, stabilizing the noisy process of mini-batch gradient descent. To our knowledge, the method was not formally proposed in any paper, but is likely inspired by the Polyak–Ruppert averaging [53,54], and is similar in essence to snapshot ensembles [55] and mean teacher [56].

Fixing the train–test resolution discrepancy: Touvron et al. [57] observe that the data augmentations that are routinely employed while training neural networks, specifically random crops, introduce a scale discrepancy. Since augmentations are usually only applied during training, the data distribution during training is skewed towards larger object sizes than during testing. They propose a simple fix, which is to first train at a smaller resolution than the target resolution, then freeze most of the weights and fine-tune only the last few layers for a couple of epochs at the target resolution. They showed that their approach reaches higher accuracy than starting training at the target resolution. Since most of the training is conducted at a lower resolution and we need much less memory for the gradients at the target resolution, we can train larger models at higher resolutions and faster training speeds on the same hardware. The ideal input resolution depends on the pixels per object in the target data, but we find empirically for our training data that doubling the training resolution performs best on average in case this information is not known.

Test-time augmentations: Another optional feature already implemented by YOLOv5 is test-time augmentations, which are an ensemble of predictions on augmented views of the same input image, such as crops and flips. They can significantly improve accuracy, especially for data with a large distribution shift compared to the training data. They are used for most state-of-the-art methods on COCO, and we similarly find that test-time augmentations offer a significant performance improvement, though at the cost of additional inference time. In our work, the inference time is the time to run the YOLO architecture on an image to acquire object detections.

Anti-aliasing: Zhang [58] shows that the down-sampling methods used by most convolutional neural networks, such as max-pooling, strided convolutions, or average pooling, violate the Nyquist–Shannon sampling theorem, making the networks lose out on their theoretical shift invariance. The author introduces BlurPool, a simple method that applies anti-aliasing before the down-sampling layers to better retain shift invariance, and shows the modified networks are more robust to shifted inputs. Although BlurPool does not improve the VisDrone validation performance, we find that it offers a qualitative improvement on out-of-domain data. BlurPool can be added to YOLOv5 with a single line of code using the composer library (https://github.com/mosaicml/composer, accessed on 20 January 2024) calling their composer.functional.apply_blurpool() method on the model. The composer implementation sacrifices some theoretical guarantees for increased efficiency, but does offer extensive benchmarks empirically showing the benefit of this trade-off.

4.6. Detection Speed and Emergency Stop

The precision of our object detection models alone is not sufficient to guarantee the safety of objects and humans in the work area. High accuracy must be combined with an

acceptable detection speed in the operational context. The 3DWI system can be divided into three large processing blocks: the camera and LiDAR calculations, the data transfer and communication to the 3D-Twin and CCU, and the 2D live detection, as presented in Figure 5.

First, the camera and LiDAR calculations are optimized using Numba [59] that converts typical coding functions such as list comprehensions and for-loops to faster machine code. Second, to ensure a resilient connection between the 3DWI and the 3D-Twin of IOSS, a wireless 4G connection between them is put in place, and the data rate is kept low. The communication only requires highly frequent commands and object information in text, and low-resolution, low-frame-rate snapshots of the left and overhead cameras. This data proved to fit easily within the available 10 Mbit/s between Sweden and the Netherlands. Third, most 3DWI processing time is used by the human activity detector and the generic object detector. To improve the detection speed, all YOLO models are converted to 8-bit integer (int8) precision and employ shared memory to efficiently get the image to the GPU. Additionally, a Nvidia Triton server (https://developer.nvidia.com/triton-inference-server, accessed on 20 January 2024) is used by the 3DWI system to manage the inference of both YOLO models and the communication to and from the GPU. Triton optimizes the 3DWI's inference speed for Nvidia GPUs, and provides the flexibility to load multiple models in memory.

The complete system must be fast enough to detect a threat when the crane is moving at its maximum set slewing speed so that the emergency stop can be initiated and completed in time. In the experiment of Section 5.3 , the FPS (frames per second) of the 3DWI system is evaluated in real-time conditions at different inference loads, i.e., running multiple detectors simultaneously. Section 5.4 will evaluate the 3DWI's performance to detect and trigger an emergency stop while the crane is slewing at its set speed. Both experiments hold that when the crane detects and stops sooner, the remaining distance between the crane's jib and the person becomes larger, and the safety is increased. We define the safety d_{safety} as the distance that remains between the red alert and the stopped crane.

$$d_{safety} = d_{observable} - d_{moved} \tag{1}$$

where $d_{observable}$ (Equation (2)) is the horizontal distance (in meters on the test terrain) that the left camera can record with halved-FOV θ, and d_{moved} is the distance traveled by the crane's spreader (and its payload) during one iteration of 3DWI processing (Equation (4)):

$$d_{observable} = tan(\theta) \cdot R_{hor} \tag{2}$$

with R_{hor} as the horizontal radius of the crane for a jib angle ϕ (upward position of the jib) and jib length L:

$$R_{hor} = cos(\phi) \cdot L \tag{3}$$

The distance d_{moved} is calculated with the delay time between red alert detection and the moment the crane comes to a full stop t_{delay} and with the horizontal linear velocity v (Equation (5)) of the crane's jib top. The linear velocity uses the angular velocity (in rad/s) of the crane as defined by Equation (6), where ω_{RPM} is the velocity given to the crane for a container movement.

$$d_{moved} = \frac{v}{t_{delay}} \tag{4}$$

$$v = \omega \cdot R_{hor} \tag{5}$$

$$\omega = 2\pi \cdot \frac{\omega_{RPM}}{60} \tag{6}$$

4.7. Detection Tracking

The results from the human activity detector and generic object detector are time-agnostic, i.e., the detector looks at a single frame at a time. Without further processing, this

may result in missing bounding boxes and size variations of bounding boxes in consecutive frames. Consistency can be improved over time using a bounding box tracker that can linearly predict bounding box locations and adapt the bounding box size of new predictions to prevent fluctuations. The tracker is applied directly after the object detector, as shown in Figure 5, and under the hood, each detector has its individual tracker.

The tracker merges detections in consecutive frames that belong to the same object in a single track. A track is "started" for detection in the current frame if it has a detection close to it in the previous frame. A track is "stopped" when the next two frames do not have a detection close to the current track detection. The bounding boxes of the tracks are converted to 3D detections used for the red alert analysis and sent to the 3D-Twin. The start of a track introduces a latency of one frame (1/FPS s).

The tracker reduces spurious detections that could issue a false alarm, provides more consistent detections, and costs one frame of latency. The increased consistency improves the robustness of the x-y-z position estimation for the tracked objects, as we use the detection boxes to select 3D coordinates based on the LiDAR measurements for the position estimation, and especially benefits the operator view in the 3D-Twin subsystem since objects will follow a smoother trajectory. The effect of the tracker is evaluated for the rooftop dataset in Section 5.5.

4.8. Jib-Top Camera Analysis

The jib-top camera as part of 3DWI's sensor suite enhances the safe crane operation. It captures parts of the environment that the oblique stereo camera and LiDARs cannot capture due to occlusions behind containers and other obstacles. It also serves as a redundancy check for the areas that are visible by the other sensors, making the system less prone to environmental effects such as sun glares. Two examples of the jib-top camera in action during the pilot demonstration are shown in Section 5.6 for qualitative inspection.

5. Results

This section describes the conducted experiments. For the comparison of AI-detection scores, the average precision AP and AP50 measures are used, as in [24,45]. The computation speed is reported as frames per second (FPS).

5.1. Experiment 1: Model Training for VisDrone

Table 1 shows the results for our first experiment, where we train and evaluate our human activity detection method (Tuned Baseline) on the VisDrone dataset. We train a YOLOv5-L model at a resolution of 1280 × 1280 with inference at 2048p while keeping the original image's aspect ratio. This means that the image from 3DWI is automatically scaled from 2560 × 2048 to 2048 × 2048, with zero padding on the top and bottom of the image. We compare to the following methods:

- Baseline model YOLOv5-L without further tuning, trained at 1280 × 1280, evaluated at 2048p.
- YOLOv7-Drone [51], trained and evaluated at 1280 × 1280.
- TPH-YOLOv5 [25] trained at 1536 × 1536, evaluated at 1996p.
- PP-YOLOE-SOD [26], to our knowledge the current state-of-the-art method, trained at 1600 × 1600, evaluated at 1920p.

As the optimal train and inference resolutions are highly dependent on the architecture and augmentation strategy, the settings selected by the original authors of each method are used. The methods are compared when trained with and without the use of ignore regions. Additionally, the baseline model and our tuned baseline model are applied for comparison to the tuned dataset (cf. Section 4.3). The results are not directly comparable to those of VisDrone because modified classes were used. The FPS is measured during inference on a single RTX 3090 video card using the inference resolutions specified above.

Table 1. Experiment 1 results. (*) The YOLOv7-Drone and PP-YOLOE-SOD measures are copied from their respective papers.

	No Ignore Regions		Using Ignore Regions		
VisDrone dataset	**AP**	**AP50**	**AP**	**AP50**	**FPS**
Baseline	0.371	0.588	0.376	0.599	14.42
YOLOv7-Drone	-	-	0.409	0.635	-
TPH-YOLOv5	**0.407**	0.627	0.415	0.642	2.21
PP-YOLOE-SOD *	-	-	0.433	0.667	3.03
Tuned Baseline (ours)	**0.407**	**0.635**	**0.439**	**0.673**	7.93
Tuned dataset	**AP**	**AP50**	**AP**	**AP50**	**FPS**
Baseline	-	-	0.398	0.668	14.42
TPH-YOLOv5	-	-	0.417	0.677	2.21
Tuned Baseline (ours)	-	-	**0.461**	**0.740**	7.93

Observations are that the baseline model has the highest image throughput, followed by the tuned baseline, PP-YOLOE-SOD, and TPH-YOLOv5. The tuned baseline model is slower due to the increased parameter count of the extra prediction head and the test-time augmentations. The TPH and PP models also perform this augmentation, have even more parameters in the neck, and use transformer layers, making them slower than pure convolutional architectures. The results show that the largest performance gain (in AP) for the tuned baseline comes from actually using the ignore regions provided by the VisDrone dataset, with much less gain for the TPH model, suggesting that it was already trained to deal with the label noise in the dataset. To conclude, the tuned baseline model and the tuned dataset together provide a good trade-off between inference speed (frame rate) and accurate detection.

5.2. Experiment 2: Model Evaluation on the Rooftop Dataset

The models trained for Section 5.1 are evaluated on the custom rooftop dataset captured by the oblique camera of the 3DWI system (cf. Figure 2, bottom row), the results of which are shown in Table 2. As these images are taken from a much closer distance to the target objects than the training data, the inference-time resolution of the 3DWI image is scaled to 1024p.

Table 2. Experiment 2 results. Object detection for the rooftop recordings.

	Person		Car	
	AP	**AP50**	**AP**	**AP50**
Baseline (COCO)	0.335	0.847	0.607	0.705
Baseline (VisDrone)	0.364	0.880	0.838	**0.995**
TPH-YOLOv5	0.381	0.918	0.845	**0.994**
Tuned Baseline (ours)	**0.426**	**0.936**	**0.907**	**0.995**

Observations are that the pre-trained COCO model can be applied to the oblique camera images with a decent performance, but that the detection accuracy is insufficient for persons and cars. The other models show a significantly higher performance on our rooftop recordings, with our tuned baseline model slightly outperforming the other models. These results indicate that the inclusion of drone-captured training data significantly improves the robustness of the human activity detector to new datasets and that it helps to tune the baseline method according to our suggestions. Based on these findings, the 3DWI is equipped with our tuned YOLO baseline and tuned VisDrone dataset as described in Section 4.

An added benefit of applying the models trained on drone-captured data to the oblique camera, which has more pixels on target relative to the overhead camera, is that the model

can accurately detect much smaller objects than models trained on conventional object detection datasets such as COCO. This allows us to reduce the input resolution without loss of accuracy and significantly increase the inference speed.

5.3. Experiment 3: Real-Time Processing

Table 3 shows the results of the speed test of the complete 3DWI system performing inference in three different configurations: the human activity detector on the oblique camera, the human activity with the generic object detector on the oblique camera, and the human activity with the generic object detector on both the oblique and jib-top camera. The speed test was performed in real-time with the sensor suite during the pilot demonstration. Since the objects are here again closer to the camera than in the training data, the human activity detector performs inference on images with a resolution of 640p, while the generic object detector uses a resolution 1280p.

Table 3. Experiment 3 results. The 3DWI system speed test for different object detector configurations.

3DWI Detection Models	FPS
Human activity (oblique)	9.0
Human activity (oblique) + generic object (oblique)	7.8
Human activity (oblique + jib top) + generic object (oblique)	7.0

The increase in the number of object detector models leads to a decrease in the overall system FPS , because the inference of all models is done sequentially. Furthermore, the addition of the generic object detection model results in a larger decrease of the system FPS compared to doubling the batch size of the human activity model (decrease of 1.2 FPS compared to 0.8 FPS, respectively) because the generic object detection model has both more parameters and requires a larger image resolution. Although in Section 5.2, a detection FPS of 7.93 is reached on a Nvidia RTX 3090 GPU at native resolution, the 3DWI system is able to achieve up to 9 FPS on a Nvidia RTX 3080 GPU with the same model due to the lower image resolution (640p) and the partial quantization to int8 precision. The end-to-end 3DWI frame rate of 7 FPS is found sufficiently fast for the pilot tests and emergency stops of the crane.

5.4. Experiment 4: Emergency Stop Safety

Two emergency stop tests were performed with the real crane during the pilot demonstration. The horizontal safety distance d_{safety} after a full stop is evaluated, as well as the time to detect and communicate the alert and the time to physically stop the crane. In both tests, the red alerts were successfully detected as soon as they entered the FOV of the camera, as shown by the red bounding boxes in Figure 8, and the crane came to a full stop before reaching the person. The timings in Table 4 were collected from the log files of the 3DWI and used to calculate the safety distance for these tests using the equations of Section 4.6.

Known values for the equations are as follows. The crane has a jib angle of $\phi = 61$ degrees and moves with a slewing speed of $\omega_{RPM} = 0.75$ rotations per minute (RPM). The crane's jib length is $L = 28$ m. The left camera has a horizontal halved-FOV of $\theta = 39$ degrees (on each side of the crane base). The 3DWI has a frame rate of $FPS_{3DWI} = 7$ FPS. With these parameters, the payload attached to the crane is moving at a horizontal linear velocity of $v = 1.1$ m per second (Equation (5)), and the left camera is able to observe $d_{observable} = 11$ m on each side of the crane's jib at the plane of the crane's jib top (Equation (2)).

With the detection, communication and deceleration timings taken into account, the time to stop the crane for test1 and test2 is, respectively, 3.5 and 2.57 s, the $d_{moved} = 3.85$ m and $d_{moved} = 2.83$ m, and $d_{safety} = 7.15$ m and $d_{safety} = 8.17$ m. In Figure 8, it can be seen that the person in test2 is in the camera's FOV slightly sooner and therefore has a larger safety distance.

Figure 8. Experiment 4 results. Red alert detection for two emergency stop tests. Note that Figure 6 shows a similar situation.

Table 4. Experiment 4 results. System delays during an emergency stop.

Emergency Tests	3DWI Processing Time	Communication Time	Time to Stop the Crane	Total Delay Time
Test 1	0.14 s	0.06 s	3.3 s	3.50 s
Test 2	0.14 s	0.03 s	2.4 s	2.57 s

Results show that the crane stops in time and the person is safe on both occasions. The largest delay is the deceleration time of the crane. With the 7 FPS processing, the 3DWI is an order of magnitude faster and sufficiently fast to trigger the emergency stop. This experiment was the first time an automated emergency stop was conducted with this crane. It is expected that the time to stop the crane can be further reduced and the safety further improved.

5.5. Experiment 5: Detection Tracking

The detection tracker is tested with the 3DWI system on the rooftop dataset. In (Figure 9), an example of six consecutive frames of the 3DWI dataset is shown with and without the tracker enabled. Without the tracker, the walking person is periodically lost for one or two frames, leading to a poor and unclear visualization in the 3D-Twin for the operator. With the tracker, the walking person is continuously detected and tracked in each frame, creating a smooth and consistent visualization. Note that the tracker made two successive predictions in order to fill the gap in Frames 2 and 3. The amount of allowed predictions depends on the allowed time the track is kept alive after no real detections and is a tunable parameter of the tracker. This parameter should be set in accordance with the system FPS. For this test, tracks were allowed to be kept alive for one second.

Figure 9. Experiment 5 results. Examples of six consecutive frames (from right to left) of the rooftop dataset without the tracker (**top**) and with the tracker (**bottom**) enabled. With the tracker a person (pink bounding boxes) is detected and tracked more consistently.

5.6. Experiment 6: Jib-Top Detection

During the pilot demonstration, several scenarios were tested in which one or multiple persons were visible in both the oblique and jib-top cameras. For three scenarios, the output of the human activity detection on both cameras is shown in Figure 10. In both viewpoints, the detector in the 3DWI was able to accurately detect and localize the visible persons and communicate the detections to the CCU and the 3D-Twin of IOSS. Although minor sun glares were visible in the oblique camera, the 3DWI managed to detect and recognize the persons in the test area.

The added viewpoint of the jib-top camera enables the 3DWI to monitor the safety of a larger area and avoid blind spots: In Figure 10, Scenario 1 shows a third person on the left side, Scenario 2 shows more of the container slots, and Scenario 3 shows the space between the containers. Moreover, considering that the human activity detector is applied to both the oblique and jib-top cameras, a level of redundancy is added to the system. This is particularly interesting for safety-critical detections, such as those triggering an emergency stop, as in Scenario 2.

Figure 10. Experiment 6 results. Each row illustrates a scenario in the pilot demonstration. The results of the human activity detector on the oblique image (**left**) and overhead image (**right**) are shown. In this test, the emergency stop functionality was disabled, meaning that detected persons are indicated as threats and shown as orange bounding boxes instead.

6. Discussion

The current sensor suite of the 3DWI is selected for daytime usage, but most of the methodology may also hold for nighttime processing. A sensor suite with two LiDARs and infrared cameras could still create the 3D obstacle map, YOLO detectors for infrared imagery exist, and the 3D-Twin construction stays the same. The situation awareness build-up for the operator is most likely the bottleneck for nighttime operation. Nonetheless, the automation of daytime logistics between small harbors and international harbors could already be a big leap into the future.

Furthermore, in the current system of systems, the 3DWI with its sensor suite is mounted on an electric GLE crane on a test site. It is expected that a similar setup can also be used on other types of cranes, as long as the jib does not occlude the sensor suite at the base and a jib-top camera can be mounted. The pilot tests are realistic scenarios but

do differ from the future use case on-board a vessel where inaccuracies in mooring and the influence of the sea state and wind affect the pose and positioning of the vessel and crane, the pendulation of the spreader, the handling of containers, and the observation of the 3D world. Motion sensors and motion compensation algorithms can be added to the processing pipeline of the 3DWI.

The use of the generic object detector in the 3DWI is still under debate. It is clearly less accurate than the human object detector, requires higher resolution input footage, and is therefore slower. It was included to also detect dogs, cats, bicycles, bags, etc. in the vicinity of the crane. However, there was insufficient time to test this in the pilot. The human activity detector worked quite well, but we did have to tune the inference-time resolution for each dataset because the pixels on target differ in our datasets. For the pilot, it worked to our advantage, because we could lower the resolution and increase the detection speed.

7. Conclusions

A 3D world interpreter system was presented as part of an autonomous robot container-handling system, with a special focus on safeguarding human activity within fifty meters of the crane. The 3DWI is part of a system of systems together with the crane control unit and intelligent operator support system. The methodology of the 3DWI is described, implemented, and used in preliminary rooftop recordings and final pilot tests. The AI detectors form the basis of the safe autonomous crane operation. They use a tuned YoloV5 baseline along with a tuned training dataset for human activity detection in both oblique and overhead imagery, as provided by the advanced sensor suite mounted on the crane. It is proved that our detectors reach state-of-the-art results on a popular small object detection benchmark and our own rooftop recordings. The current 3DWI processing speed of 7 FPS is sufficiently fast to issue the crane's emergency stop, and two pilot tests show that the safety distance after an emergency stop remains larger than 7, 8 m. To issue an emergency stop, a suspicious detection has to be within fifty meters of the crane.

The LiDAR and stereo sensors, dock-scan procedure, and DEM analysis facilitate the conversion of the environment scan, detected containers, and other objects to a 3D-Twin of the situation. During the pilot tests, the 3DWI in Örnsköldsvik communicated remotely over 4G with the 3D-Twin in Soesterberg 1500 km away, displaying the live movements of the crane, spreader and containers, 3D detections from the object detector, and error messages of the system to improve the situation awareness of the remote operator. With a basic interface in the 3D-Twin, the operator interacts with the 3DWI and checks the latest stereo and jib-top image for a final assessment. Then the operator releases the alerted state, and the autonomous crane continues its safe operation until the work order is completed.

In future work, the 3DWI concept can be extended for its use in the RCHS on a dynamic vessel, e.g., by using pose and positioning systems and motion compensation algorithms or by continuously matching 2D/3D features of the sensor suite with predefined features in each specific harbor. On top of that, it is interesting to research extensions of the concept towards a vessel with two collaborating cranes that provide each other with scans and alerts from a different point of view and both change the configuration of containers. Furthermore, it is important to extensively test the 3DWI method in different seasons with different weather conditions and to further improve on the human activity detection for both the oblique and top view imagery.

Author Contributions: Conceptualization, methodology, supervision, funding, F.B.t.H.; Software, formal investigation, resources, F.R. and B.T.v.M.; Validation, visualization, writing, F.B.t.H., F.R. and B.T.v.M. All authors have read and agreed to the published version of the manuscript.

Funding: This project has received funding from the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement No. 861678.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The training data for the human activity detector are from publicly available datasets and cited in the paper: VisDrone, HERIDAL, and MOCS. Rooftop recordings and pilot demonstration recordings are not readily available due to technical limitations. Requests for access to the datasets should be directed to the first author.

Acknowledgments: The authors are particularly grateful to the team of Macgregor Cranes AB in Örnsköldsvik. They provided the pilot test site, the crane and spreader, and the CCU interface, and together, we developed and tested the final pilot demonstrator. The TNO team in Soesterberg with their IOSS in the immersive collaboration lab has been of great importance to the research and demonstration. We thank the TNO team in The Hague for providing the required software toolboxes and 4G-infrastructure support. Finally, we acknowledge the Horizon2020 Moses consortium for the continuous support and review of intermediate results.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

3D-Twin	3D Digital Twin
3DWI	3D World Interpreter
AP	Average Precision
CCU	Crane Control Unit
COCO	Common Objects in Context
DEM	Digital Elevation Map
FOV	Field of View
FPS	Frames Per Second
GPU	Graphics Processing Unit
IOSS	Intelligent Operator Support System
LiDAR	Light Detection and Ranging
MOSES	autoMated vessels and supply chain Optimization for Sustainable short SEa Shipping
PTU	Pan-Tilt Unit
RCHS	Robotic Crane-Handling System
RGB	Red, Green, Blue
RPM	Rotations Per Minute
SSS	Short Sea Shipping
TNO	Netherlands Organisation for Applied Scientific Research
YOLO	You Only Look Once

References

1. Ghaderi, H. Autonomous technologies in short sea shipping: Trends, feasibility and implications. *Transp. Rev.* **2019**, *39*, 152–173. [CrossRef]
2. Tiusanen, R.; Malm, T.; Ronkainen, A. An overview of current safety requirements for autonomous machines—Review of standards. *Open Eng.* **2020**, *10*, 665–673. [CrossRef]
3. Mohseni, S.; Pitale, M.; Singh, V.; Wang, Z. Practical Solutions for Machine Learning Safety in Autonomous Vehicles. *arXiv* **2019**, arXiv:1912.09630.
4. Wang, J.; Zhang, L.; Huang, Y.; Zhao, J.; Bella, F. Safety of autonomous vehicles. *J. Adv. Transp.* **2020**, *2020*, 8867757. [CrossRef]
5. Perez-Cerrolaza, J.; Abella, J.; Borg, M.; Donzella, C.; Cerquides, J.; Cazorla, F.J.; Englund, C.; Tauber, M.; Nikolakopoulos, G.; Flores, J.L. Artificial Intelligence for Safety-Critical Systems in Industrial and Transportation Domains: A Survey. *ACM Comput. Surv.* **2023**, *Just Accepted*. [CrossRef]
6. Karvonen, H.; Heikkilä, E.; Wahlström, M. Safety Challenges of AI in Autonomous Systems Design—Solutions from Human Factors Perspective Emphasizing AI Awareness. In *Engineering Psychology and Cognitive Ergonomics. Cognition and Design*; Springer: Copenhagen, Denmark, 2020; pp. 147–160.
7. Almeaibed, S.; Al-Rubaye, S.; Tsourdos, A.; Avdelidis, N.P. Digital Twin Analysis to Promote Safety and Security in Autonomous Vehicles. *IEEE Commun. Stand. Mag.* **2021**, *5*, 40–46. [CrossRef]
8. Stączek, P.; Pizoń, J.; Danilczuk, W.; Gola, A. A Digital Twin Approach for the Improvement of an Autonomous Mobile Robots (AMR's) Operating Environment—A Case Study. *Sensors* **2021**, *21*, 7830. [CrossRef] [PubMed]

9. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012.
10. Qian, R.; Lai, X.; Li, X. 3D Object Detection for Autonomous Driving: A Survey. *Pattern Recognit.* **2022**, *130*, 108796. [CrossRef]
11. Filgueira, A.; González-Jorge, H.; Lagüela, S.; Díaz-Vilariño, L.; Arias, P. Quantifying the influence of rain in LiDAR performance. *Measurement* **2017**, *95*, 143–148. [CrossRef]
12. Jokela, M.; Kutila, M.; Pyykönen, P. Testing and Validation of Automotive Point-Cloud Sensors in Adverse Weather Conditions. *Appl. Sci.* **2019**, *9*, 2341. [CrossRef]
13. Abdo, J.; Hamblin, S.; Chen, G. Effect of Weather on the Performance of Autonomous Vehicle LiDAR Sensors. In Proceedings of the ASME International Mechanical Engineering Congress and Exposition, Virtual, 1–5 November 2021. [CrossRef]
14. Sebastian, G.; Vattem, T.; Lukic, L.; Bürgy, C.; Schumann, T. RangeWeatherNet for LiDAR-only weather and road condition classification. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; pp. 777–784. [CrossRef]
15. Kumar, D.; Muhammad, N. Object Detection in Adverse Weather for Autonomous Driving through Data Merging and YOLOv8. *Sensors* **2023**, *23*, 8471. [CrossRef]
16. Qu, S.; Yang, X.; Zhou, H.; Xie, Y. Improved YOLOv5-based for small traffic sign detection under complex weather. *Sci. Rep.* **2023**, *13*, 16219. [CrossRef]
17. Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L.; Shum, H.Y. DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection. In Proceedings of the The Eleventh International Conference on Learning Representations, Virtual, 25–29 April 2022.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015.
19. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the CVPR, Salt Lake City, UT, USA, 18–22 June 2018.
20. Woo, S.; Debnath, S.; Hu, R.; Chen, X.; Liu, Z.; Kweon, I.S.; Xie, S. ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders. *arXiv* **2023**, arXiv:2301.00808.
21. Zong, Z.; Song, G.; Liu, Y. DETRs with Collaborative Hybrid Assignments Training. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 4–6 October 2023.
22. Liu, Z.; Hu, H.; Lin, Y.; Yao, Z.; Xie, Z.; Wei, Y.; Ning, J.; Cao, Y.; Zhang, Z.; Dong, L.; et al. Swin Transformer V2: Scaling Up Capacity and Resolution. In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022.
23. Vo, N.D.; Nguyen, L.; Ngo, G.; Du, D.; Do, L.; Nguyen, K. Transformer-based End-to-End Object Detection in Aerial Images. *Int. J. Adv. Comput. Sci. Appl.* **2023**, *14*, 1072–1079. [CrossRef]
24. Cao, Y.; He, Z.; Wang, L.; Wang, W.; Yuan, Y.; Zhang, D.; Zhang, J.; Zhu, P.; Van Gool, L.; Han, J.; et al. VisDrone-DET2021: The vision meets drone object detection challenge results. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 2847–2854.
25. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-Captured Scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788.
26. Xu, S.; Wang, X.; Lv, W.; Chang, Q.; Cui, C.; Deng, K.; Wang, G.; Dang, Q.; Wei, S.; Du, Y.; et al. PP-YOLOE: An evolved version of YOLO. *arXiv* **2022**, arXiv:2203.16250.
27. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
28. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
29. Jocher, G. Software implementation YOLOv5 by Ultralytics. Available online: https://zenodo.org/records/7347926 (accessed on 20 January 2024).
30. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.
31. Golcarenarenji, G.; Martinez-Alpiste, I.; Wang, Q.; Alcaraz-Calero, J.M. Machine-learning-based top-view safety monitoring of ground workforce on complex industrial sites. *Neural Comput. Appl.* **2022**, *34*, 4207–4220. [CrossRef]
32. Sutjaritvorakul, T.; Vierling, A.; Pawlak, J.; Berns, K. Simulation platform for crane visibility safety assistance. In *Advances in Service and Industrial Robotics: Results of RAAD*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 22–29.
33. Sutjaritvorakul, T.; Vierling, A.; Berns, K. Data-driven worker detection from load-view crane camera. In Proceedings of the International Symposium on Automation and Robotics in Construction, Online, 27–28 October 2020; IAARC Publications: Lyon, France, 2020; Volume 37, pp. 864–871.
34. Neuhausen, M.; Herbers, P.; König, M. Using synthetic data to improve and evaluate the tracking performance of construction workers on site. *Appl. Sci.* **2020**, *10*, 4948. [CrossRef]
35. He, T.; Zhang, Z.; Zhang, H.; Zhang, Z.; Xie, J.; Li, M. Bag of tricks for image classification with convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 558–567.

36. Zhang, Z.; He, T.; Zhang, H.; Zhang, Z.; Xie, J.; Li, M. Bag of freebies for training object detection neural networks. *arXiv* **2019**, arXiv:1902.04103.
37. Steiner, A.; Kolesnikov, A.; Zhai, X.; Wightman, R.; Uszkoreit, J.; Beyer, L. How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers. *arXiv* **2021**, arXiv:2106.10270.
38. Yang, B.; Luo, W.; Urtasun, R. PIXOR: Real-time 3D Object Detection from Point Clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
39. Arikumar, K.S.; Deepak Kumar, A.; Gadekallu, T.R.; Prathiba, S.B.; Tamilarasi, K. Real-Time 3D Object Detection and Classification in Autonomous Driving Environment Using 3D LiDAR and Camera Sensors. *Electronics* **2022**, *11*, 4203. [CrossRef]
40. Middelhoek, F. Stereo Pointclouds for Safety Monitoring of Port Environments. Master's Thesis, TUDelft, Delft, The Netherlands, 2023.
41. Zhu, P.; Wen, L.; Du, D.; Bian, X.; Fan, H.; Hu, Q.; Ling, H. Detection and Tracking Meet Drones Challenge. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 7380–7399. [CrossRef]
42. Božić-Štulić, D.; Marušić, Ž.; Gotovac, S. Deep learning approach in aerial imagery for supporting land search and rescue missions. *Int. J. Comput. Vis.* **2019**, *127*, 1256–1278. [CrossRef]
43. Xuehui, A.; Li, Z.; Zuguang, L.; Chengzhi, W.; Pengfei, L.; Zhiwei, L. Dataset and benchmark for detecting moving objects in construction sites. *Autom. Constr.* **2021**, *122*, 103482. [CrossRef]
44. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
45. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.
46. Micikevicius, P.; Narang, S.; Alben, J.; Diamos, G.; Elsen, E.; Garcia, D.; Ginsburg, B.; Houston, M.; Kuchaiev, O.; Venkatesh, G.; et al. Mixed Precision Training. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
47. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond Empirical Risk Minimization. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
48. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 390–391.
49. Moore, B.E.; Corso, J.J. FiftyOne. GitHub. 2020. Available online: https://github.com/voxel51/fiftyone (accessed on 20 January 2024).
50. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020.
51. Fu, X.; Wei, G.; Yuan, X.; Liang, Y.; Bo, Y. Efficient YOLOv7-Drone: An Enhanced Object Detection Approach for Drone Aerial Imagery. *Drones* **2023**, *7*, 616. [CrossRef]
52. Northcutt, C.G.; Athalye, A.; Mueller, J. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. In Proceedings of the Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1), Virtual, 7–10 December 2021 .
53. Polyak, B.T. New stochastic approximation type procedures. *Automat. Telemekh* **1990**, *7*, 2.
54. Ruppert, D. *Efficient Estimators from a Slowly Convergent Robbins-Monro Procedure*; Cornell University Operations Research and Industrial Engineering: Ithaca, NY, USA, 1988; Volume 781.
55. Huang, G.; Li, Y.; Pleiss, G.; Liu, Z.; Hopcroft, J.E.; Weinberger, K.Q. Snapshot Ensembles: Train 1, Get M for Free. *arXiv* **2017**, arXiv:1704.00109.
56. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1195–1204.
57. Touvron, H.; Vedaldi, A.; Douze, M.; Jégou, H. Fixing the train-test resolution discrepancy. *Adv. Neural Inf. Process. Syst. (NeurIPS)* **2019**, *32*.
58. Zhang, R. Making Convolutional Networks Shift-Invariant Again. In Proceedings of the ICML, Long Beach, CA, USA, 9–15 June 2019.
59. Numba: A LLVM-based Python JIT Compiler. In Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC, Austin, TX, USA, 15–20 November 2015; pp. 1–6.