Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/EU08.20250000
Published: July 2, 2025

A Digital Twin for Air Cargo Ground Operations: Connecting the Real with the Virtual

Jenny Tonka and Michaël Schyns
Center for Quantitative Methods and Operations Management
HEC Liège - Management School of the University of Liège
Liège, Belgium
j.tonka@uliege.be, m.schyns@uliege.be

Abstract

This paper proposes a comprehensive digital twin design for monitoring and optimizing air cargo ground operations, i.e., the logistical activities that take place between the time a cargo aircraft lands and takes off again. Specifically, we decided to focus on gathering the real-time data required to build the essential real-to-virtual connection of the digital twin, that would ultimately feed an optimization algorithm. This is a real challenge given the highly constrained and regulated environment of airports, where traditional data collection techniques, such as information systems and Internet of Things, have proven insufficient for our data needs. We therefore developed a computer vision-based approach that leverages airports' existing network of cameras to identify and track ground service vehicles, allowing real-time monitoring of cargo ground operations and aircraft (un)loading progress. Technically, we collected real and synthetic visual data of air cargo ground operations and studied different labeling strategies to overcome the challenges of occlusion, resource shape variation, and detection stability. We then trained the pre-trained YOLO11n object detection model on the generated labeled datasets and used the BoT-SORT tracking algorithm. The results obtained are promising. In particular, we achieved an overall mAP50-95 of 0.883 (resp. 0.929) on real (resp. synthetic) data. This supports the idea that computer vision is a good approach to connect the real with the virtual in the context of a digital twin for air cargo ground operations monitoring and optimization. A future research direction would be to improve the method further by designing a tailored tracking algorithm.

Keywords
Digital twin, Computer vision, Air cargo ground operations.

1. Introduction
A digital twin is a virtual replication of a physical entity (and its environment) that is entirely synchronized with its counterpart thanks to the combination of a physical-to-virtual connection with a virtual-to-physical one (Tonka and Schyns 2021). The concept was first introduced by Michael Grieves in 2002 during a special meeting on product life-cycle management (Jones et al. 2020; Rathore et al. 2021). Since then, digital twins have attracted the interest of many researchers and professionals. This is mainly due to the closed loop created by the continuous bidirectional data flow between physical and virtual parts. This indeed leads to a real-time monitoring of the physical entity, which cannot be achieved by traditional modeling methods (Rathore et al. 2021). In addition, the virtual-to-physical connection, that does not exist in conventional simulation exercises, enables digital twins to test and subsequently adjust virtual hypotheses based on physical feedback, creating a continuous optimization cycle (Jones et al. 2020). This tool is also very versatile, it can be used in numerous domains, from manufacturing to education, transportation, or medicine (Rathore et al. 2021), and can be leveraged for many purposes, such as real-time monitoring or process optimization (Tonka and Schyns 2021).

1.1 Problem Description

In this paper, we will focus on the design of a digital twin for the monitoring and optimization of air cargo ground operations. These refer to the logistical activities that are performed between the time a cargo aircraft lands and takes off again, and mainly include fueling, catering, cleaning, technical checks, and, obviously, cargo handling. As can be seen in Figure 1, they require the use of many different types of ground service vehicles, which must be coordinated in the most efficient way according to a number of synchronization constraints. In particular, ground operations presented in parallel in Figure 1 can be performed simultaneously, while those in series should be conducted sequentially due to precedence constraints. There also exist goods transfers between (un)loading vehicles that affect the start, operating, and end times of all the vehicles involved, as well as unpowered vehicles that should be paired with powered ones to move in space. Air cargo ground operations are in addition the most critical airport processes in terms of flight delays (Padrón and Guimarans 2019). Yet, with the continued growth of e-commerce, in which air transport plays a major role, there is a pressing demand to limit delays as much as possible. Indeed, beyond their negative impact on client satisfaction, flight delays result in additional operation costs for airlines (Britto et al. 2012) and may also cause additional environmental damage by increasing fuel consumption and gas emissions (Ryerson et al. 2014). We therefore strongly believe that creating a digital twin to monitor and optimize air cargo ground operations could be a solution. We have already created a virtual optimization algorithm in Tonka et al. (2024), where we developed a client-centered heuristic approach using a recursive procedure that is able to create ground service vehicle routes such that all synchronization constraints are met and the total service time is minimized, ultimately helping to reduce the number and duration of delays. The next step in the digital twin design is the gathering of real-time data about ground operations and service vehicles. This will allow us to closely monitor air cargo ground operations in real time, feed the optimization algorithm with the required data, and efficiently respond to any deviations from the optimal plan, closing the loop between the physical and virtual parts of our digital twin.

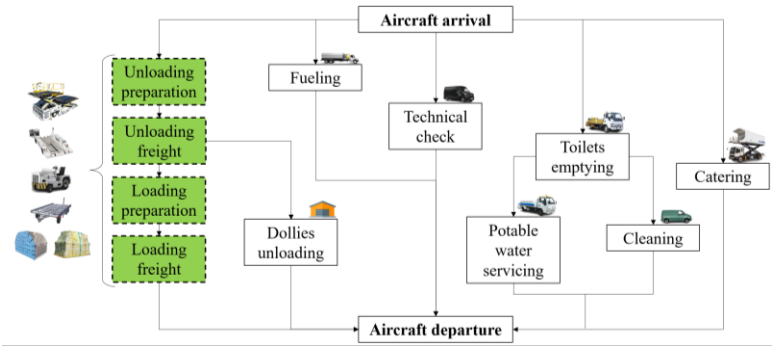


Figure 1. Air cargo ground operations diagram

1.2 Data Collection Strategy

A major challenge in achieving this goal is determining the strategy for collecting the real-time data we seek. Indeed, airports are highly specific and regulated environments where traditional data collection techniques, such as information systems and Internet of Things (IoT), have proven insufficient for our data needs. For instance, while airports have robust internal information systems that track flight departures and arrivals, they usually lack information on operation status and vehicle locations, which are essential for closely monitoring air cargo ground operations. When looking at internal information systems of handlers and airlines, although they may contain valuable information, these are fragmented systems that may not be compatible. Merging and standardizing data would therefore be a time-consuming process, which is not consistent with our goal of real-time monitoring. In addition, there is no guarantee that all the necessary data will be available. Indeed, handlers and airlines may be unwilling or unable to share their information due to airport security regulations, confidentiality concerns, or legal and ethical considerations, especially in light of the General Data Protection Regulation (GDPR). It is also possible that the information we are looking for simply does not exist in their systems. Next, in the search for additional data, IoT seems like a natural approach. We therefore considered a GPS-based solution, deploying forty sensors on ground service vehicles at Liège Airport, one of the main cargo airports in Europe, to track their real-time location. While this solution has potential, the results of our experiment show that several challenges are still to be overcome. First, GPS sensors are primarily designed for outdoor use, which limits their effectiveness for tracking vehicles in indoor facilities such as warehouses. Second, the harsh operating conditions of airports can affect the reliability of sensor fasteners, resulting in the loss of two sensors during the experiment. Then, maintaining battery life, especially for unpowered vehicles, is a significant logistical challenge. In addition, detailed progress monitoring of (un)loading operations is limited by the complexity of

managing GPS sensors for all unit load devices. Finally, GPS sensors often lack the accuracy required. For example, the GPS sensors tested at Liège Airport have an average accuracy of 20 meters, with performance ranging from 4 to 210 meters. It is even possible that some movements were not recorded at all. While satisfactory compromises can be found to address previous problems, this lack of accuracy makes IoT systems unsuitable for the detailed monitoring goal we are pursuing. Therefore, given these limitations, we decided to turn to machine learning and artificial intelligence solutions. Since airports are typically equipped with a network of cameras, we decided to leverage this existing infrastructure. We developed a computer vision-based approach that uses deep learning architectures to extract information from images in the same way that human vision would do (Alam et al. 2021). We believe that it is a promising solution for real-time data collection in complex environments such as airports. This is supported by Yıldız et al. (2022), who claim that a human-oriented system for monitoring airport ground operations would compromise data reliability and be too labor-intensive.

1.3 Objectives

Our main objective and contribution to the scientific literature lies in the comprehensive design of a digital twin for monitoring and optimizing air cargo ground operations, taking into account all the constraints and regulations related to the field. In particular, this paper focuses on the real-to-virtual connection part of the digital twin, that would provide the necessary data to the previously developed optimization algorithm (Tonka et al. 2024), aiming to reduce flight delays and their associated negative impact on client satisfaction, airline operating costs, and the environment. In this context, we have conducted an analysis of machine learning techniques to design a computer vision-based system that is able to both identify and track key resources over time, while overcoming the challenges of resource similarity, occlusion, and shape variations, inherent in cargo ground operations. We specifically designed a detailed monitoring of aircraft (un)loading progress, that would help to provide real-time estimates of the operation end time and identify any deviations from the optimal plan. Beyond addressing a managerial question, we therefore also contribute to the technical side of moving from theory to practice, especially regarding data preprocessing.

The remainder of this paper is organized as follows: Section 2 provides a literature review on computer vision for airport ground operations, Section 3 details the methods, and Section 4 presents the results of our experiments. Section 5 finally discusses future research directions and draws conclusions.

2. Literature Review

As mentioned above, ground operations are one of the most critical processes in airports and are therefore one of the main causes of flight delays. We believe that a digital twin dedicated to the close monitoring and optimization of these operations can reduce the number and duration of delays, and thereby all the costs associated with them. To collect the required data, we decided to adopt a computer vision-based approach, and it seems that the scientific literature is also moving in this direction. Indeed, computer vision is increasingly being considered for use in the air industry, it is already popular for detecting aircraft and people at airports, but so far little research has been done on ground service vehicles (Ding et al. 2023). Among the few papers that already exist, we identify two categories: those that only consider object detection, and those that also consider tracking.

Considering the works of the first group, interesting ones are those of Ding et al. (2023) and Xu et al. (2024), both of which have developed specific surveillance algorithm for ground operations. Still, these are not clearly in line with what we intend to do. On the one hand, Xu et al. (2024) chose to annotate the operations themselves rather than the vehicles involved when training their object detection model. In contrast, we decided to focus on ground service vehicles. Indeed, some operations, such as cargo (un)loading, are very complex and involve the use of multiple vehicles. While the individual presence of these vehicles has little meaning from an operational point of view, it is an important piece of information for resource management. On the other hand, Ding et al. (2023) focused on pixel-level detection to capture the exact appearance and irregular shape of ground service vehicles, considering that this would allow for a higher detection accuracy. Our research will however show later that detection methods using bounding boxes are accurate enough to monitor ground operations closely.

Next, looking at the second category of papers, it is interesting to consider the works of Van Phat et al. (2020, 2022) and Yıldız et al. (2022). Van Phat et al. (2020, 2022) both consider the monitoring of ground operations through computer vision, and combine the detection of ground service vehicles with their role in ground operations to predict push-back times. Although these studies are close to ours in that they have management insights, as does our digital twin, they have some limitations. For instance, Van Phat et al. (2020) focus only on the operations that contribute to

the push-back time prediction, meaning that they do not consider ground operations global monitoring as we intend to do. The same conclusion can be drawn for the level of detail considered: while Van Phat et al. (2020) are satisfied with information on operations beginning and end, we aim to achieve a higher level of detail that allows monitoring operations progress (e.g. counting in real time the number of containers and pallets that have left/entered a given aircraft). This, in turn, will make it possible to estimate the end time of operations in real time and identify any deviations from the optimal plan. Regarding Yıldız et al. (2022), they designed a system that automatically detects and tracks ground service vehicles in real time to monitor ground operations. This is getting very close to what we intend to do, but an important difference is to be underlined. Indeed, as most of the literature, the focus of Yıldız et al. (2022) is put on the passenger side of the air industry. In contrast, we focus on the cargo side of the air industry. Although this may seem insignificant, the specificities of cargo ground operations lead to a higher level of complexity in terms of ground service vehicles detection and tracking. Indeed, cargo ground operations involve specific vehicles that have very similar shapes, such as speed loaders and dollies, or that change shape over time, such as high loaders (see Figure 2). In addition, the number of occlusions increases significantly because most vehicles involved in air cargo ground operations are already partially occluded by their own loads.

In conclusion, while computer vision for ground operations monitoring is promising, it is still in its early stages. Our contribution is therefore to propose a detailed and global monitoring approach specifically tailored to the complexities of cargo ground operations. Note that additional references on the techniques used to identify and track resources are provided in Section 3.

3. Methods

Given our objective of tracking air cargo ground service vehicles to design the real-to-virtual connection of our digital twin, we need to track the movements of multiple vehicles in real time. This is referred to as multi-object tracking. According to Aharon et al. (2022) and Zhang et al. (2022), the most effective method for multi-object tracking is tracking-by-detection, that is a method that includes an object detection step followed by a tracking step.

3.1 Labeled Dataset Creation

Computer vision, like any other deep learning model, requires data on which to train. To this end, although the access to visual data of air cargo ground operations was limited, we collected several videos and extracted individual frames from them. This process yielded a dataset of 2057 images with different weather and lighting conditions, as we used videos recorded on sunny days, clear nights, and rainy nights. In addition, the collected videos come from different airports and airlines, resulting in a wide variety of equipment shapes and appearances. However, due to the limited size of the dataset, this diversity results in only a few instances of each vehicle variant. Although we will use a transfer learning approach that partially addresses this problem (see Section 3.2), it should be noted that this may be a challenge for training the object detection model.

Beyond images, computer vision requires labeled images. Image labeling involves annotating images by identifying objects of interest, assigning them category labels, and drawing bounding boxes to define their precise locations. Manually labeling thousands of images is a significant task, especially since a single image can contain multiple instances (i.e., labeled objects). We have therefore defined some guidelines to ensure consistency throughout the whole process and to address the challenges posed by the specific context of air cargo ground operations. First, to avoid introducing bias in the object detection model through training on objects that are not clearly defined, we decided not to label vehicles that are too blurred, too small, or too far away to be easily detected by the human eye. This mainly concerns vehicles that are outside of the aircraft parking stand. Then the first major challenge we face is the many occlusions that ground service vehicles are exposed to. We have defined three rules to address this. First, vehicles that are partially occluded by their own load are labeled regardless of the percentage of occlusion. This makes sense, since cargo (un)loading ground service vehicles are loaded most of the time. Second, for vehicles that are partially out of the camera field and vehicles that are occluded by the aircraft, another vehicle, or any other object, only vehicles that are at least 50% visible are labeled. Cargo aircraft parking stands are indeed often crowded areas as many operations are performed in a short period of time and, given the similarities that exist between some cargo ground service vehicles, we consider that vehicles that are less than 50% visible are too occluded to be recognized with 100% certainty even by the human eye. Third, occluded objects that are labeled are given bounding boxes as if they were fully visible. This allows to increase the tracking stability for vehicles that become partially occluded during the monitoring process. Next, since our digital twin is dedicated to air cargo ground operations, we decided to focus on the labeling of (un)loading vehicles and categorize them as shown in Figure 2. (Un)loading operations, shown in green in Figure 1,

are the most critical part of air cargo ground operations as they require the coordination of many different vehicle types. We therefore assume that the development of an efficient object detection model and tracking algorithm for these operations would be easily generalizable to the entire process afterwards. Looking at (un)loading vehicles, high loaders, used as lifts between the ground and aircraft cargo doors, pose a second significant object detection challenge due to their shape variations during the process. To ease detection, we decided to assign different labels to high loaders depending on their shape. However, this increases tracking complexity since a single resource is assigned multiple labels. Finally, the labeled dataset has been split into training (70%), validation (20%), and test (10%) sets, ensuring an equal proportion of vehicle types in each. Two additional videos were also collected to evaluate the performance of the models in terms of detection and tracking stability, which is the third major challenge we have to address.

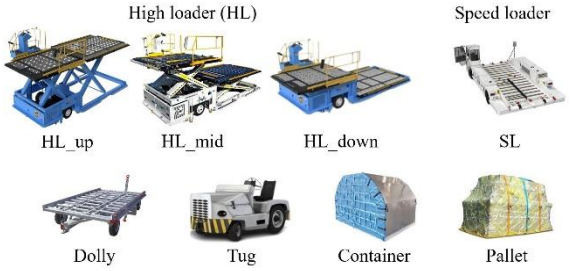


Figure 2. Labeling of (un)loading ground service vehicles

3.2 Object Detection Model

In the literature, there exist several object detection techniques. Voulodimos et al. (2018) show that the most common ones are Convolutional Neural Networks (CNNs), Stacked Denoising Autoencoders (SDA), Deep Belief Networks (DBNs), and Deep Boltzmann Machines (DBMs). However, CNNs and their variants seem to be the preferred deep learning architecture for computer vision (Alam et al. 2021; Voulodimos et al. 2018). This is explained by the fact that CNNs are able to achieve good detection accuracies, mainly due to their invariance to transformations (scaling, rotation, or translation) and their ability to automatically learn features based on the given dataset (Voulodimos et al. 2018). A large part of the literature focuses specifically on Region-based Convolutional Neural Networks (R-CNNs), a method that uses a region proposal method to first generate potential bounding boxes and then trains CNNs to classify the proposed boxes into object categories or background (Redmon et al. 2016; Ren et al. 2017). Yet, CNNs and R-CNNs are computationally expensive to train (Voulodimos et al. 2018). Much research has therefore been devoted to improving the performance of R-CNNs. In this regard, Ren et al. (2017) propose an object detection system called Faster R-CNN. It consists of two modules, one is a CNN that proposes regions, while the other is the Fast R-CNN detector (Girshick 2015) that uses the proposed regions (Ren et al. 2017). Still, although the computational burden of R-CNNs has been drastically reduced over time (Ren et al. 2017), they remain slow (Redmon et al. 2016).

In this paper, we chose to use the YOLO object detection algorithm. This model has been introduced by Redmon et al. (2016) and uses a single CNN to simultaneously predict multiple bounding boxes and class probabilities for those boxes. Unlike R-CNNs, which are two-step processes, YOLO stands for You Only Look Once at an image, meaning that it trains directly on full images and is therefore able to detect objects in a in a single step (Kaur et al. 2024; Redmon et al. 2016). The main advantage of this property is that YOLO is extremely fast, being able to reach real-time speed while maintaining more than twice the mean average precision of other real-time systems (Redmon et al. 2016). In addition, we opted for pre-trained YOLO models that initialize their parameters based on the extensive MS COCO (Microsoft Common Objects in Context) dataset. The MS COCO dataset contains 2.5 million labeled instances in 328,000 images (Lin et al. 2014), providing a solid foundation for training computer vision models and enabling the use of transfer learning. Beyond accelerating the learning process and enhancing the generalization ability of the network (Voulodimos et al. 2018), transfer learning helps to solve the problem of insufficient training data by transferring the knowledge from the source domain to the target domain (Tan et al. 2018). Since data collection in domains such as the air cargo industry is very complex, and since deep learning requires massive training data in order to achieve good performance (Tan et al. 2018), the ability of pre-trained YOLO models to generalize from small datasets (Sharma et al. 2024) makes them very suitable for our use case.

Several versions of YOLO have been released since its inception. Kaur et al. (2024) present the framework, architecture, performance, and added features of most YOLO versions, from the original to YOLOv8. Each

advancement is the result of modifications intended either to improve performance in terms of speed and accuracy, or to address model limitations (Kaur et al. 2024). It therefore appears that YOLO and its derivatives are pushing the state-of-the-art in terms of accuracy, speed, and memory usage on numerous benchmarks (Kaur et al. 2024). This is confirmed by the work of Bhavya Sree et al. (2021), where YOLO leads to the best accuracy-speed tradeoff among R-CNN, Fast R-CNN, Faster R-CNN, and Single Shot MutliBox Detector (SSD). Sharma et al. (2024) show that these conclusions are still valid for the latest versions of YOLO (from YOLOv8 to YOLO11). It has also been shown that YOLO models are particularly good at discriminating between vehicle types (Kim et al. 2020). These results therefore further support our decision to use YOLO for object detection. In particular, we decided to work with YOLO11. Beyond being the latest version of YOLO, YOLO11 proves to achieve competitive accuracy with faster inference than its predecessors (Sharma et al. 2024). It moreover comes with different model sizes. Given our real-time application, we opted for the smallest model (YOLO11n) as deepening the model improves its precision but slows down the detection speed (Xu et al. 2024).

3.3 Tracking Algorithm

According to Aharon et al. (2022), the current state-of-the-art tracking algorithm is BoT-SORT. The latter is a modified and improved version of the ByteTrack algorithm (Aharon et al. 2022). ByteTrack already ranked first on both MOT17 and MOT20 (which contains numerous crowded scenarios and occlusion cases) at the time of its release and achieved state-of-the-art performance on the HiEve and BDD100K tracking benchmarks (Zhang et al. 2022). This was mainly due to its robustness to occlusion, which results from the fact that, unlike other tracking methods that only retain high score detection boxes, ByteTrack also considers low score boxes, which are typically associated with occlusion, motion blur, or size-changing situations (Zhang et al. 2022). The modifications built into ByteTrack to create the BoT-SORT algorithm further improved this performance on the different tracking benchmarks, outperforming all other popular trackers on all main multi-object tracking metrics (Aharon et al. 2022). Given the context of air cargo ground operations, where both tracking accuracy and occlusion management are of utmost importance, we decided to proceed with the BoT-SORT tracking algorithm.

4. Results and Discussion

We ran our experiments on a Windows 11 laptop equipped with an AMD Ryzen 9 5900HX CPU, 32 GB of RAM, and an NVIDIA GeForce RTX 3070 GPU. Since we have chosen a tracking-by-detection method for the tracking of air cargo ground service vehicles, we need to design an efficient object detection model before moving on to the actual tracking. Object detection models are typically evaluated using the precision, recall, and mean Average Precision (mAP) metrics: Precision refers to the accuracy of detection and is equal to the proportion of correct detections out of the total number of detections made by the model; Recall measures the ability of the model to detect target objects and is equal to the proportion of correct detections out of all relevant objects to be detected in the dataset; The mAP computes the average area under the precision-recall curve across multiple object classes and provides an overall measure of performance at different Intersection over Union (IoU) thresholds (Sharma et al. 2024). The IoU refers to the overlap between the predicted and actual bounding boxes, we report mAP50 (50% IoU) to assess performance on easy detections and mAP50-95 (50% to 95% IoU) for a global evaluation across different levels of detection difficulty.

4.1 Object Detection Model - Real Data

We configured the training process of the pre-trained YOLO11n model with a total of 1000 training epochs to obtain a model as accurate as possible. The numerous tests that we have performed indeed suggest that 1000 training epochs is high enough to ensure the convergence of the model in the specific context of ground operations monitoring. However, to prevent overfitting, YOLO set a default patience value of 100 to stop training after 100 epochs with no improvement in the validation metrics.

After 537 epochs, the model achieved its best performance with an overall mAP50 of 0.972 and mAP50-95 of 0.883 on the test set. Detailed results, reported in Table 1, show a high overall precision of 95.9% and individual precisions that do not go below 91.8%. The recall is also good, with an average across all classes of 93.7%, but is slightly below the precision, with some individual recalls going below 90%. This is the case for dollies and speed loaders, which are visually similar and often occluded. Tests on unseen videos confirm the good performance of the detection model as shown in Figure 3, but also suggest a lack of detection stability. In particular, very few vehicles are (correctly) detected when occlusion is important, or when vehicles are in a crowded area. Yet, given the approach we have chosen, a lack of detection stability in real-time video could subsequently prevent tracking stability. Therefore, while these results are already promising, we aimed to further improve the performance metrics to ensure robust and accurate vehicle

detection. To achieve this, we focused on refining the training dataset. Since no additional data was available, we decided to experiment with different labeling strategies.

Table 1. Original detection model results on the test set

Class	Images	Instances	Precision	Recall	mAP50	mAP50-95
all	205	1517	0.959	0.937	0.972	0.883
Dolly	136	482	0.979	0.886	0.961	0.755
HL_down	122	125	0.985	0.992	0.995	0.987
HL_mid	24	24	0.918	0.937	0.958	0.951
HL_up	32	32	0.95	0.969	0.992	0.992
Pallet	122	409	0.979	0.971	0.987	0.859
SL	15	17	0.92	0.882	0.926	0.768
Tug	122	151	0.966	0.935	0.978	0.878
Container	128	277	0.974	0.928	0.975	0.872

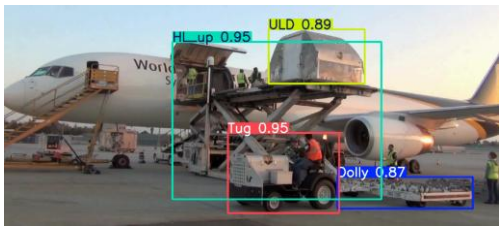


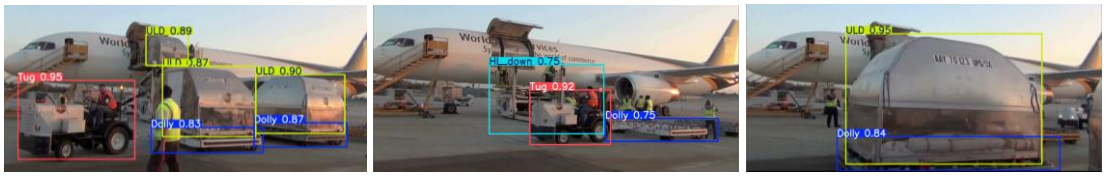
Figure 3. Original detection model results on an unseen video

A first hypothesis was that eliminating the labeling of vehicles that are blurry, have lighting problems, or are too far away might improve detection stability. Therefore, we modified the original dataset to focus only on perfectly visible vehicles in the foreground, raising up our previous minimum quality requirements. The pre-trained YOLO11n model was trained on this modified dataset using the same training configuration as before. The model achieved its best performance at epoch 261, with a mAP50 of 0.838 and a mAP50-95 of 0.796 on the original test set. Detailed results are reported in Table 2 and show that the drop in the mAP metrics is due to a significant lack of recall, suggesting that training on the foreground only will not improve detection stability. Indeed, although the model achieves an overall precision of 95.3%, with individual precisions fairly similar to the previous ones, the overall recall reaches only 74%. Looking at the individual metrics, it seems that the model especially fails to detect dollies, pallets, and containers, which are often stored all over the airport ground and therefore in more distant or crowded areas. Tugs, which may also operate in such areas, see their individual recall reduced as well, but to a lesser extent, while other classes of vehicles, which are mostly used directly in the aircraft parking stand, have similar or improved individual recalls compared to before. Therefore, while stricter labeling may improve performance and provide the desired detection stability under ideal foreground conditions, it may not be the optimal approach for air cargo ground operations, where challenging conditions are often encountered (vehicles may be distant, in a crowded area, partially occluded, or affected by adverse weather conditions and varying lighting). This is confirmed by tests on unseen videos, where the model appears to struggle at the first signs of occlusion, as shown in Figure 4. We will therefore stick to a labeling style that considers both foreground and background resources.

Table 2. Foreground labeling detection model results on the original test set

Class	Images	Instances	Precision	Recall	mAP50	mAP50-95
all	205	1517	0.953	0.74	0.838	0.796
Dolly	136	482	0.976	0.409	0.608	0.539
HL_down	122	125	0.992	0.973	0.985	0.977
HL_mid	24	24	0.929	0.958	0.986	0.973
HL_up	32	32	0.95	1	0.99	0.988
Pallet	122	409	0.981	0.372	0.622	0.578
SL	15	17	0.881	0.875	0.938	0.849

Tug	122	151	0.947	0.848	0.904	0.832
Container	128	277	0.966	0.484	0.669	0.633



Original detection model

Foreground labeling detection model

Figure 4. Difference between detection model results on an unseen video

Second, although we initially decided to assign different labels to high loaders because their shapes vary a lot during (un)loading operations, we hypothesized that simplifying the labeling process by assigning a single label (HL) to high loaders, regardless of their state, might improve detection stability. To test this hypothesis, we modified the original dataset and trained the pre-trained YOLO11n model on it using the same training configuration as before. This new model achieved a mAP50 of 0.968 and a mAP50-95 of 0.85 on the test set, reaching its best performance at epoch 466. Detailed results are reported in Table 3. The model achieves a high overall precision of 96.1% (with individual precision not falling below 91.7%), while the overall recall reaches 93.5% (with some individuals going below 90%). Compared to the original model, the results are very similar. The observed variations are indeed so minimal that equivalent performance is obtained when testing the model on unseen videos. Therefore, simplifying the labeling for high loaders seems to have a negligible impact on the overall stability of the detection model.

Table 3. Simplified labeling detection model results on the test set

Class	**Images**	**Instances**	**Precision**	**Recall**	**mAP50**	**mAP50-95**
all	205	1517	0.961	0.935	0.968	0.85
Dolly	136	482	0.97	0.876	0.949	0.747
HL	178	181	0.983	0.994	0.995	0.987
Pallet	122	409	0.972	0.971	0.986	0.863
SL	15	17	0.917	0.882	0.929	0.753
Tug	122	151	0.96	0.962	0.976	0.877
Container	128	277	0.964	0.924	0.975	0.871

Despite the strong performance of our models, we believe that further refinement of the dataset could yield improvements, especially in terms of detection stability, which is a key element for tracking-by-detection approaches as ours. Indeed, the size of our real data training set remains limited and, as already mentioned, the variety of airports and airlines considered leads to few instances of each vehicle variant. Therefore, given the difficulty of collecting additional real-world data, we decided to use a detailed virtual environment of Liège Airport to create a synthetic dataset. This approach would help minimize variations in vehicle shapes and appearances, allowing for the collection of a larger number of instances for each specific vehicle variant, which we believe would increase detection stability.

4.2 Object Detection Model - Synthetic Data

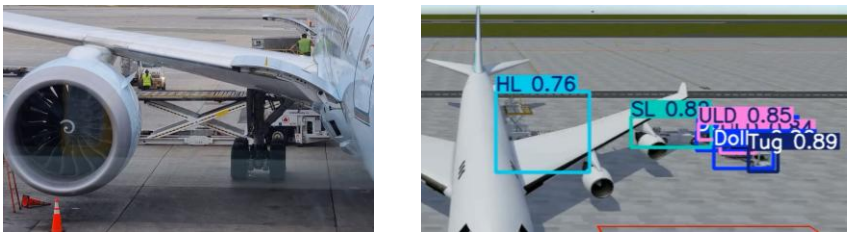
To build the virtual entity and interface of our digital twin, our lab developed a complete 3D model of Liège Airport, one of the main cargo airports in Europe. This virtual environment not only replicates a real airport, but also includes the modeling of various ground service vehicles and resources, allowing the simulation of any ground operation. The virtual environment of Liège Airport was therefore used to create our synthetic training dataset, recording virtual videos of various scenarios. By extracting individual frames, we created a dataset of 3990 images, almost twice the

size of the original real-world dataset, with 70% used for training, 20% for validation, and 10% for testing. Again, we ensure the same proportion of vehicle types in the three datasets. Image labeling was automated using a Unity script, following the same labeling rules as the original real-world dataset. Note, however, that in order to simplify subsequent tracking and since both labeling styles seem to lead to equivalent results, we assigned a single label to high loaders.

We trained the YOLO11n pre-trained model using the same configuration as before (1000 training epochs and a patience value of 100). The model achieved its best performance at epoch 381, with a mAP50 of 0.993 and a mAP50-95 of 0.929 on the test set. Detailed results, reported in Table 4, show an overall precision of 98.3% and an overall recall of 98%, with individual performance not falling below 95.5% and 96.5%, respectively (tugs even reach 100% recall). This improved performance, especially in terms of recall, compared to the model trained on real data with simplified labeling, should lead to higher detection stability. This is confirmed by tests on unseen videos from the virtual environment. In particular, it appears that the synthetic data detection model outperforms the real data detection model when it comes to occlusion. Indeed, as can be seen in Figure 5, the real data detection model fails to detect a high loader hidden behind an aircraft wing while the one trained on synthetic data succeeds.

Table 4. Synthetic data detection model results on the test set

Class	Images	Instances	Precision	Recall	mAP50	mAP50-95
All	415	1402	0.983	0.98	0.993	0.929
Dolly	187	312	0.99	0.965	0.994	0.959
HL	280	280	0.996	0.996	0.994	0.983
Pallet	92	130	0.977	0.984	0.993	0.831
SL	157	157	0.955	0.968	0.99	0.958
Tug	189	189	0.999	1	0.995	0.945
Container	224	334	0.982	0.967	0.989	0.902



Simplified labeling detection model Synthetic data detection model

Figure 5. Difference between detection model results for occlusion cases

Given these good results and knowing that the virtual environment in which we collected our dataset is very representative of reality, we investigated whether training on our virtual dataset could produce good results when tested on real images. However, it appears that direct transfer to real-world scenarios was not successful. The model indeed showed poor performance on real images, with all metrics approaching zero for all vehicle types. We then investigated whether training on a combined dataset consisting of both real and virtual images in a 50/50 ratio would improve the metrics and stability of detection on real images. A hybrid dataset was created consisting of the real dataset with simplified labeling and half of the synthetic dataset. However, the results were comparable, if not worse, than those obtained with the simplified labeling real dataset alone, indicating that the inclusion of the virtual dataset did not improve performance on real data. Nevertheless, the improved performance and higher detection stability demonstrated by the detection model trained on the synthetic dataset makes it a valuable tool for developing and refining tracking algorithms. Indeed, given the limitations of real-world data availability that constrain detection stability, if tracking errors occur, the model trained on the synthetic dataset would help to determine whether they are due to instability in resource detection or to the intrinsic quality of the tracking algorithm. This distinction is essential for evaluating and improving the performance of the tracking algorithm.

4.3 Tracking Algorithm

Besides identifying ground service vehicles, our digital twin requires real-time monitoring of air cargo ground operations, and therefore accurate tracking of these vehicles. Our approach is to track ground service vehicles as they

enter and exit an aircraft parking stand, so that we can monitor the start and end times of each ground operation on that aircraft. In addition, we want to monitor the progress of (un)loading operations by counting containers and pallets that enter and exit an aircraft in real time. This would make it possible to provide real-time estimates of the operation end time and identify any deviations from the optimal plan.

As mentioned before, we decided to use the BoT-SORT tracking algorithm. Regarding the parameterization, we used an argument which tells the tracker that the current frame is the next in a sequence and that it should expect details of the previous frame in the current one. After some tests on unseen videos, we set the minimum confidence threshold for the detection model trained on real (synthetic) data equal to 75% (70%), as this value leads to the best results in terms of detection accuracy and stability, while avoiding too many false positives.

We tested this tracking method on unseen real videos, using the detection model previously trained on the real dataset with simplified labeling. We chose this detection model so that only one tracking ID is assigned to the high loader (and we already showed that this labeling style does not affect the detection model's performance). As expected, the lack of stability of the detection model leads to many tracking errors. In fact, because the detection model is not stable enough, achieving tracking stability is challenging. Some identifications are lost and then found again, but the associated tracking ID does not remain the same, except when the identification loss is very short in time. For example, in Figure 6, we can see that within a few seconds of operations, the identification of the second unloaded container changes from ID 13 to ID 18, while the identification of the high loader is lost.

Figure 6. Tracking errors for real data simplified labeling detection model

Given these results, we decided to use the model trained on synthetic data to test the tracking method presented above in a context of higher detection stability. As expected, the tracking performance is significantly improved. We can accurately capture the arrival/departure times of ground service vehicles at the aircraft parking stand, allowing the construction of a monitoring dashboard. We are also able to monitor aircraft (un)loading progress, counting in real time the number of containers and pallets that have left/entered a given aircraft. For instance, in Figure 7, there is one and then two containers that are unloaded from an aircraft, with the counter in the upper right increasing accordingly. This confirms that a very high level of detection stability is required for the tracking-by-detection method we consider. However, we already achieve a very high level of performance with our detection models trained on real datasets, so it seems unrealistic to achieve even higher performance. To overcome this limitation, a solution would be to turn away from classic tracking algorithms to design a tailored one, which we leave for future work. Still, we believe that this approach is the most suitable one for designing the real-to-virtual connection we need to build the monitoring and optimizing digital twin of air cargo ground operations that we are looking for.

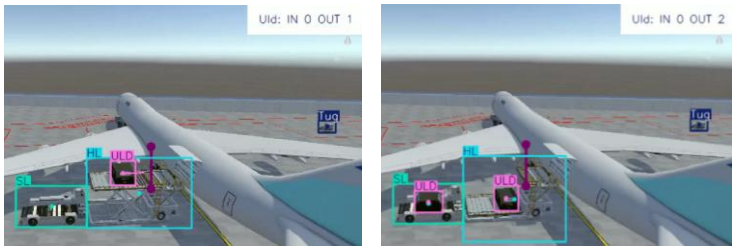


Figure 7. Aircraft unloading progress monitoring

5. Conclusion

The objective of this research is to provide a comprehensive design of a digital twin for air cargo ground operations. Specifically, we focus on gathering the data required to build the essential real-to-virtual connection of the digital

twin, that would ultimately feed the optimization algorithm developed in a previous work (Tonka et al. 2024). We indeed demonstrated in Tonka et al. (2024) that with accurate real-time data available, ground operations can be significantly improved, leading to drastic reductions in flight delays. This is however a real challenge given the highly constrained and regulated environment of airports. In particular, traditional data collection techniques, such as information systems and IoT solutions, have proven insufficient for our data needs. We therefore turned to computer vision, leveraging airports' existing network of cameras to identify and track ground service vehicles. Although this approach is increasingly being considered in the literature, little research has been done so far. Among the few existing papers, our work differs from others in that it considers cargo ground operations rather than passenger ones, and provides monitoring at a high level of detail, allowing to track in real-time the progress of aircraft (un)loading operations. In addition, different labelling strategies are studied and tested to overcome the challenges of occlusion, resource shape variation, and detection stability. Beyond addressing a managerial question, we therefore also contribute to the definition of an appropriate way to preprocess training data in the context of computer vision.

Technically, we collected real-world videos of air cargo ground operations, defined labeling guidelines, and created several variants of a labeled dataset. We then identified the pre-trained YOLO11n object detection model and the BoT-SORT tracking algorithm as the most suitable for our research. Although the object detection models trained on our different datasets showed good performance metrics, testing on unseen videos suggested a lack of tracking stability, leading to identification loss and changes over time. We suspected that this was due to a lack of stability in the object detection itself, since it is a key element of the tracking-by-detection method we use. We therefore used a detailed virtual environment of Liège Airport to generate a synthetic dataset and train a new object detection model that showed a higher detection stability. Testing our tracking algorithm again, the results were much improved. In particular, we were able to collect the necessary data to build a ground operations monitoring dashboard with detailed information on aircraft (un)loading progress. These results lead us to believe that computer vision is a good approach for designing the real-to-virtual connection of a digital twin for air cargo ground operations optimization. However, it is still necessary to achieve comparable detection stability for real-world data. While additional refinements to the training dataset could marginally improve the performance of object detection models, it is unrealistic to aim for 100% precision and recall across all vehicle types. An interesting future work would therefore be to address this limitation by designing a tailored tracking algorithm. A comprehensive review of the handling of challenging scenarios such as poor lighting, adverse weather, and occlusion would also be of great interest for future research.

Acknowledgements

Jenny Tonka is a Research Fellow of the Fonds de la Recherche Scientifique – FNRS

References

Aharon, N., Orfaig, R. and Bobrovsky, B. Z., BoT-SORT: Robust Associations Multi-Pedestrian Tracking, *arXiv preprint*, 2022.

Alam, E., Sufian, A., Das, A. K., Bhattacharya, A., Ali, M. F. and Rahman, M. M. H., Leveraging deep learning for computer vision: A review, *2021 22nd International Arab Conference on Information Technology (ACIT)*, pp. 1-8, Muscat, Oman, December 21-23, 2021.

Bhavya Sree, B., Yashwanth Bharadwaj, V. and Neelima, N., An Inter-Comparative Survey on State-of-the-Art Detectors—R-CNN, YOLO, and SSD, *Intelligent Manufacturing and Energy Sustainability (ICIMES)*, pp. 475-483, Hyderabad, India, August 21-22, 2021.

Britto, R., Dresner, M. and Voltes, A., The impact of flight delays on passenger demand and societal welfare, *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 2, pp. 460-469, 2012.

Ding, M., Zhou, W., Xu, Y. and Xu, Y., Two-stage Framework for Specialty Vehicles Detection and Classification: Toward Intelligent Visual Surveillance of Airport Surface, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 2, pp. 1912-1923, 2023.

Girshick, R., Fast R-CNN, *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440-1448, Santiago, Chile, December 7-13, 2015.

Jones, D., Snider, C., Nassehi, A., Yon, J. and Hicks, B., Characterising the Digital Twin: A systematic literature review, *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36-52, 2020.

Kaur, S., Kaur, L. and Lal, M., A Review: YOLO and Its Advancements, *Proceedings of International Conference on Recent Innovations in Computing (ICRIC)*, vol. 2, pp. 577-592, Budapest, Hungary, December 21-22, 2024.

Kim, J. A., Sung, J. Y. and Park, S. H., Comparison of Faster-RCNN, YOLO, and SSD for Real-Time Vehicle Type Recognition, *2020 IEEE International Conference on Consumer Electronics – Asia (ICCE-Asia)*, pp. 1-4, Seoul, South Korea, November 1-3, 2020.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L., Microsoft COCO: Common Objects in Context, *13th European Conference on Computer Vision (ECCV)*, pp. 740-755, Zurich, Switzerland, September 6-12, 2014.

Padrón, S. and Guimarans, D., An Improved Method for Scheduling Aircraft Ground Handling Operations From a Global Perspective, *Asia-Pacific Journal of Operational Research*, vol. 36, no. 4, pp. 1950020, 2019.

Rathore, M. M., Shah, S. A., Shukla, D., Bentafat, E. and Bakiras, S., The Role of AI, Machine Learning, and Big Data in Digital Twinning: A Systematic Literature Review, Challenges, and Opportunities, *IEEE Access*, vol. 9, pp. 32030-32052, 2021.

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., You Only Look Once: Unified, Real-Time Object Detection, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788, Las Vegas, USA, June 27-30, 2016.

Ren, S., He, K., Girshick, R. and Sun, J., Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 2017.

Ryerson, M. S., Hansen, M. and Bonn, J., Time to burn: Flight delay, terminal efficiency, and fuel consumption in the National Airspace System, *Transportation Research Part A: Policy and Practice*, vol. 69, pp. 286-298, 2014.

Sharma, A., Kumar, V. and Longchamps, L., Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species, *Smart Agricultural Technology*, vol. 9, pp. 100648, 2024.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C. and Liu, C., A Survey on Deep Transfer Learning, *27th International Conference on Artificial Neural Networks (ICANN)*, pp. 270-279, Rhodes, Greece, October 5-7, 2018.

Tonka, J., Paquay, C. and Schyns, M., A Recursive Algorithm for Synchronized Rich Vehicle Routing in Air Cargo Ground Handling, *Manuscript submitted for publication*, 2024.

Tonka, J. and Schyns, M., The digital twin concept: A definition attempt, *ORBi-University of Liège*, 2021.

Van Phat, T., Alam, S., Lilith, N. and Nguyen, B. T., A computer vision framework using Convolutional Neural Networks for airport-airside surveillance, *Transportation Research Part C: Emerging Technologies*, vol. 137, pp. 103590, 2022.

Van Phat, T., Alam, S., Lilith, N., Tran, P. N. and Nguyen, B. T., Aircraft Push-back Prediction and Turnaround Monitoring by Vision-based Object Detection and Activity Identification, *10th SESAR Innovation Days*, Online, December 7-10, 2020.

Voulodimos, A., Doulamis, N., Doulamis, A. and Protopapadakis, E., Deep Learning for Computer Vision: A Brief Review, *Computational Intelligence and Neuroscience*, vol. 2018, no. 1, pp. 7068349, 2018.

Xu, Y., Liu, Y., Shi, K., Wang, X., Li, Y., Chen, J., Xu, Y., Liu, Y., Shi, K., Wang, X., Li, Y. and Chen, J., An airport apron ground service surveillance algorithm based on improved YOLO network, *Electronic Research Archive*, vol. 32, no. 5, pp. 3569-3587, 2024.

Yıldız, S., Aydemir, O., Memiş, A. and Varlı, S., A turnaround control system to automatically detect and monitor the time stamps of ground service actions in airports: A deep learning and computer vision based approach, *Engineering Applications of Artificial Intelligence*, vol. 114, pp. 105032, 2022.

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W. and Wang, X., ByteTrack: Multi-object Tracking by Associating Every Detection Box, *2022 17th European Conference on Computer Vision (ECCV)*, pp. 1-21, Tel Aviv, Israel, October 23-27, 2022.

Biographies

Jenny Tonka graduated from HEC Liège with a Master's Degree in Business Engineering with a specialization in Supply Chain Management and Business Analytics in 2021. She is a PhD candidate working in the QuantOM (Quantitative methods and Operations Management) research center of HEC Liège. Her doctoral research is related to the use of digital twins in the air cargo industry and is performed under the supervision of Prof. Michaël Schyns.

Michael Schyns holds degrees in Computer Science and in Management Sciences. He is a full professor of Digital Business at HEC Liège – Management School of the University of Liège. His main fields of interest are Business Analytics, Machine Learning, Operations Research, and new digital technologies for business. He is the head of a research and development lab in Augmented and Virtual Reality.