AN UNMANNED AERIAL SYSTEM FOR ON-THE-EDGE TRAFFIC MONITORING IN WORK-ZONE APPROACH AREA

J Martínez-Sánchez [1]*, D. Conde Morales [1], S. Novoa [1], P. Arias [1]

[1] Applied Geotechnology Group, CINTECX, Universidade de Vigo, 36310 Vigo, Spain - (joaquin.martinez, david.conde.morales, samuel.novoa, parias)@uvigo.gal

KEY WORDS: Edge-computing, UAS, Work Zone monitoring; Optical sensors, GPU processing

ABSTRACT:

Maintenance works are crucial to improve the reliability and resilience of road infrastructure but, despite efforts to achieve a safer operation, work zones are still risky areas where 4% of all road accidents occur. The main factors that increase the risk during maintenance include the proximity to live traffic, inadequate warning signs and driver behaviour. Intelligent Transportation Systems and their supporting technologies including sensors, data processing and analysis have been beneficial for increasing road safety.
In this work, we present the design of a context awareness approach based on an Unmanned Aerial System aimed to detect inadequate speed of incoming traffic approaching to a work zone and to raise warning alerts. To accomplish this objective, an optical payload carrying an on-the-edge analysis system based on deep learning tracking was developed and tested. Preliminary results show the potential of the design to achieve near real-time operation preserving a mean Average Precision similar to that obtained with more complex architectures.

1. INTRODUCTION

Improving the reliability and resilience of road infrastructure throughout its lifecycle is of great importance. To achieve this, road maintenance has been improving through a paradigm shift from corrective methods to predictive ones (Rúa et al., 2022). Despite efforts to improve road safety, work zones remain a priority as 4% of all accidents occur in these areas, that entail significant risks to both road users and workers (European Transport Safety Council, 2011). The execution of repetitive tasks increases the likelihood of errors and, therefore, associated risks. This is one of the reasons why there is a growing interest in the use of infrastructure maintenance robots to improve efficiency and safety.

On the other hand, intelligent transportation systems (ITS) have made significant progress in recent decades with benefits in sustainability and safety improvements. ITS are multidisciplinary systems that combine Information and Computing Technologies (ICT), including sensors, data processing and analysis methods, and communications, to support conventional transport infrastructure (Lin et al., 2017). In the construction field, optical sensors that allow for monitoring and mapping of the scene enable real-time supervision of construction activities, both indoors and outdoors (Rao et al., 2022). These improvements in ITS are useful for monitoring the traffic approaching work zones on roads during maintenance.

In this manuscript, we present the design of a context awareness and monitoring approach based on an Unmanned Aerial System (UAS). The objective of this work is to define an intelligent UAS-payload for on-the-edge processing and analysis of video frames to detect incoming traffic to the work-zone and to raise alerts in case of imminent risk.

2. PROPOSED SOLUTION

The proposed solution consists of the following components:

- Unmanned Aerial Vehicle (UAV) platform
- Optical payload and image analytics

2.1 UAV Platform

In a UAS, the UAV platform is the first subsystem to be defined, because the operation and regulations depend on its characteristics.

In our case, the main aim of the UAS is the monitoring of the different space segments of the infrastructure where a Work Zone is defined. These spaces consist of the *approach area*, the *queuing area* , the inner work zone and the *termination area* (Dirección General de Carreteras, 2014). Figure 1 depicts the approaching areas before reaching the inner WorkZone, that are the target areas for UAS-based monitoring.

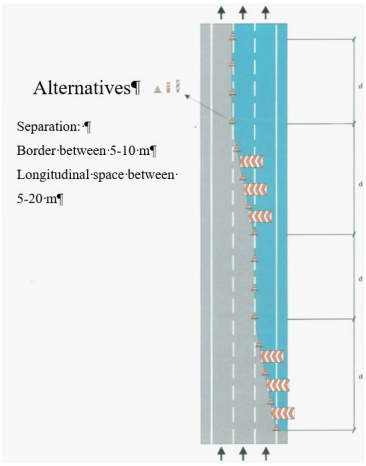

Figure 1 Segmentation of WorkZone in Approach Area, Queuing Area, Inner WorkZone.

* Corresponding author

Taking into account the main objectives of the UAS, the most suitable UAV platform is a rotary wing vehicle, also known as multicopter.

The most salient feature of this platform consists of its capability to hover on the air, thus maintaining the position that is collected by means of a GNSS component. In order to improve the precision of the positioning, a Real-time Kinematic (RTK) solution is required.

The main drawback of these platforms is that the autonomy of the system is low compared to fixed wing platform. To overcome this limitation, the platform is powered by a ground station that is tethered to the UAV through a cable. As a result, the ground station provides both power and control capabilities to the system that can perform the context awareness functions with no restrictions of time (Elistair | The Tethered Drone Company, 2023). Figure 2 shows the UAV flying with the tethering cable attached to the Ground Power system.

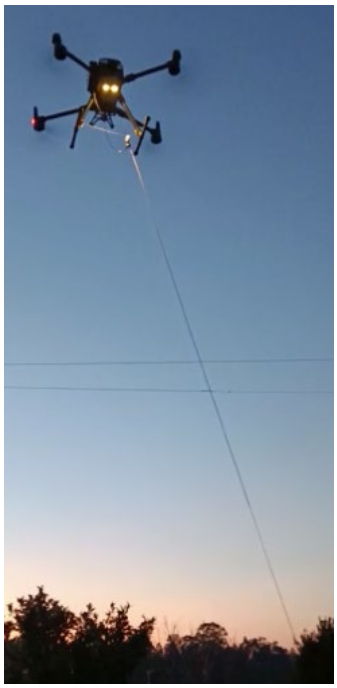

Figure 2 UAV Ground-powered with a tethering system

2.2 Optical payload

The optical system of the UAS consists of a RGB camera mounted with a zoom lens that allows for traffic monitoring in the *approach area* of the Working Zone. The distance to the Working Zone is defined to alert users and personnel with a sufficient time to respond to the traffic risk, and in our case consist of 400 m, that is the distance from the starting point of the *Approaching Area* (view Figure 1).

2.3 Image analytics

The analytics of the context awareness system consists of a Multi Object Tracking (MOT) algorithm based on ByteTrack including (i) a Deep Learning based vehicle detector, (ii) an instance tracker based on Kalman filtering, and (iii) an analysis module to provide traffic statistics and raise alerts in case of risk. Similarly, ByteTrack has gained attention because of its tracking capabilities based on the combination of object detection and tracking into a unified framework. It integrates the benefits of both tracking-by-detection (in frame) and tracking-by-regression (in consecutive frames) approaches, making it robust and accurate in real-time handling of complex tracking scenarios (Zhang et al., 2022). The workflow of the image analysis is shown in Figure 3.

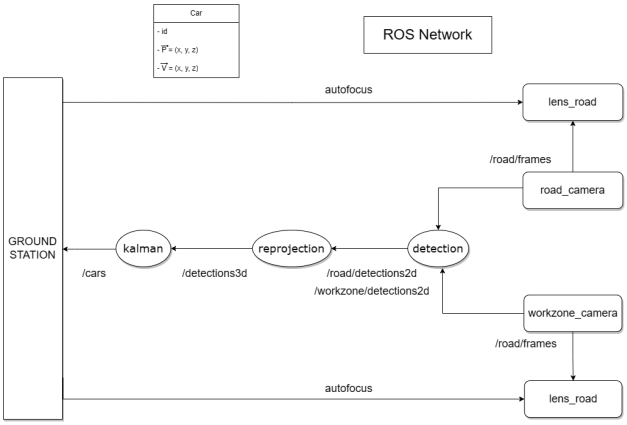

Figure 3 Software workflow design

2.3.1 Deep learning vehicle detector. The state of the art for image detection consists of the advanced full deep neural networks (DNNs) and, among the proposed architectures, YOLO (You Only Look Once) has emerged as a highly efficient detection framework. The architecture of YOLO consists of a neural network that is divided into two parts: a feature extractor, typically a pre-trained convolutional neural network (CNN), and a detection head based on convolutional and fully connected layers (Redmon et al., 2016). As a result, YOLO takes an input image and outputs a set of bounding boxes and class probabilities for the objects detected in the image. In particular, YOLO-v8 has shown performance improvements and, considering that v8 is focused on hardware efficiency and architectural reforms, this results in in better throughput with a similar number of parameters compared to previous versions of the detector (Hussain, 2023).

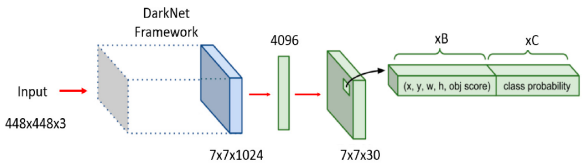

Figure 4 Preliminary architecture of YOLOv1 as in (Hussain, 2023)

The training and validation of the Neural Network was conducted on non-pretrained models using different combinations of real and synthetic datasets obtained from Nvidia's Omniverse platform (Conde et al., 2023).

2.3.2 Reprojection of detection points on the ground. In order to obtain real-world measurements from the bounding-box vehicle detections, we use the reprojection of the image coordinates of a reference point onto a photogrammetric 3D model that was created using images collected with a UAV surveying.
To derive the reference points from the detections, the following assumptions are made:

1. Vehicles can be approximated to a rectangular prism in shape, so the overall structure of vehicles tends to resemble a rectangular base, with the body extending on height. This approximation helps visualize the general dimensions of vehicles and the space they occupy on the road.
2. The road is locally flat, meaning that there are no significant variations in height in the immediate surroundings of the car. This indicates that the road terrain remains locally even and consistent.
3. We can exploit the fact that, considering that the actual dimensions of the vehicle are not of interest to us, and the curvature ratios in high-capacity roads we can simplify the problem by assuming that the object under study maintains the same rotation with respect to the camera. By doing so, we achieve a centre local invariance with respect to the object's rotation.

Considering that (Xl, Yl) and (Xu, Yu) are the corners of the bounding box, we obtain the coordinates of the centre of the upper and lower segments:

Once these coordinates on the image are known, we will use the intrinsic parameters and distortion coefficients to obtain the direction vector of the line that passes through the points in the world space that correspond to that pixel in image space. This method aims to correct lens distortion, but it also returns the normalized coordinates, and, therefore, we can construct the desired direction vector using the resulting X and Y components obtained from the execution of this method, by setting the Z component equal to 1.
Once this vector is known, it is rotated using the inverse rotation vector extracted from the camera's position to transform it into world space and, using a lightweight library, we perform raycasting on the 3D model of the zone, obtaining the three-dimensional point where the object's bounding box intersects with the ground for the central point of the lower segment. Regarding the upper point and since we have assumed that the road is locally flat, the raycasting is not performed directly on the model itself. Instead, it is performed on a plane positioned at the same height as the lower point of the bounding box to form a right triangle behind the car.
Considering the positioning of these 3D points and the timestamp of the video frames, we can calculate both the trajectory and speed of the vehicles to be tracked on the images. An extended Kalman filter is used to filter the trajectory.

2.4 On-the-edge processing and implementation

The algorithm runs in a GPU-based embedded processing system that aims to provide real-time monitoring of the traffic flow.

DNNs can be quite computationally extensive, comprising multiple layers and a multitude of weights. As a consequence, DNNs demand significant computing power, memory, and energy usage, which can pose challenges for embedded devices with limited capabilities, making it difficult to deploy and run the models efficiently. However, there are optimized inferencers, such as TensorRT, that can leverage the capabilities of open-source frameworks and enable low-latency and high-throughput execution on embedded systems by re-implementing and deploying pre-trained DNNs.

The optimization of the inference engine involves two steps: compilation and runtime optimization. During the compilation phase, the inference engine makes decisions on how to execute the layers and perform the operations efficiently. One optimization technique involves weight and activation precision calibration, where the engine reduces the precision from FP64 to FP32 or even quantizes them to INT8 whenever feasible. By reducing the precision, the engine can achieve a good trade-off between computational accuracy and efficiency. Additionally, layer and tensor fusion techniques are applied to minimize the number of operations required while maintaining precision. By fusing multiple layers into a single operation, the engine can reduce the computational overhead and exploit the GPU memory and bandwidth more effectively. This fusion process ensures that the model's precision is minimally affected while maximizing the utilization of available resources.

In the runtime phase, the inference engine focuses on executing the network as efficiently as possible. Integration with popular frameworks provides the opportunity to replace generic layer implementations with those specifically optimized for NVIDIA graphics devices. By embedding executors designed for the specific hardware, the inference engine can leverage device-specific optimizations and hardware acceleration, thereby improving the overall performance and efficiency of the inference process.

This combination of compilation and runtime optimization techniques enables the inference engine to deliver high-performance and resource-efficient execution of deep neural networks, unlocking a higher potential of NVIDIA graphics devices for deep learning applications.

3. RESULTS AND DISCUSSION

3.1 Location and Description of the test site

The test site for the designing activities described in this document is the AG41 road in Pontevedra, a province in northwest Spain. AG41 plays a crucial role in facilitating seamless connectivity and transportation efficiency between a touristic area and the major AP9 highway. As a high-capacity road, it is designed to handle significant traffic volume, ensuring smooth flow and reducing congestion. The specific location consists of the Kilometric Points (PK) 12 -13 of the road, depicted in Figure 5.

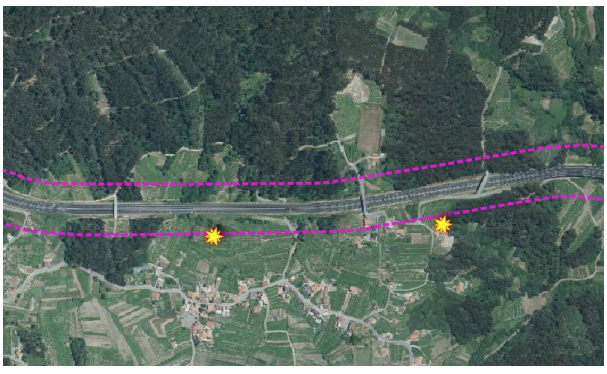

Figure 5 Aerial view of the test site in PK12-13 of the AG41

3.2 Image analytics results

In order to obtain the performance of object detection modules, we calculated the mean Average Precision (mAP), a common metric for this purpose that is derived at an Intersection over Union (IoU) threshold. This means that, to calculate mAP@50, we need the Average Precision (AP) for each of the classes in our dataset. We obtain this by inspecting the area under the precision-recall curve (view Figure 6).

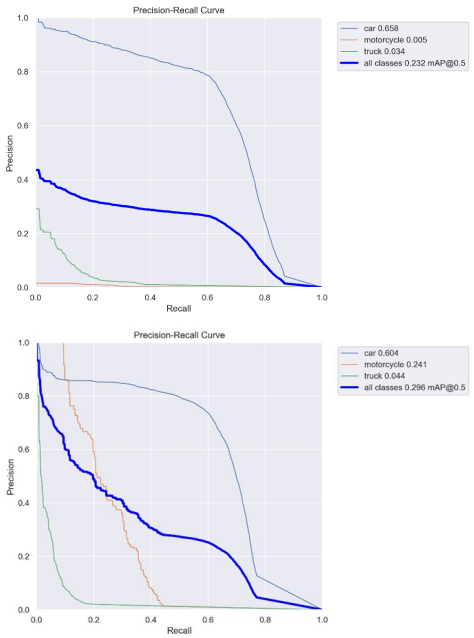

Figure 6 Precision-Recall curve for the training with (a) only real data and (b) including synthetic data

It is important to mention the effect of using synthetic data in the training and validation dataset in this experiment that was beneficial, as it resulted in an improvement in the mean average precision while also reducing the variation between different classes. Particularly noteworthy was the ability to successfully detect the motorcycle class, which is typically more challenging to identify. This achievement was possible by incorporating synthetic data alongside the limited number of real data samples. By augmenting the dataset with synthetic examples, the model's performance was enhanced, leading to a higher overall average precision and a notable reduction in the variability across different classes.

Focusing on mAP metrics, mAP50 is depicted in the following table along with the more stringent evaluation mAP50-95, that shows average mAP over different IoU thresholds, from 0.5 to 0.95.

	mAP50	mAP50-95
Car	0.570	0.342
Motorcycle	0.206	0.082
Truck	0.040	0.023
All	0.272	0.149

Table 1 mAP Values

These validation metrics before the optimization were obtained with Ultralytics YOLOv8 and a custom dataset of 3770 images. Figure 7 shows the output for this stage of the methodology.

Figure 7 Example of bounding box vehicle detections in the test site.

By applying the procedure to reproject the coordinates of the derived locally invariant centre of the detection bounding boxes on the ground, and the timestamp between frame, instantaneous speed is calculated (view Figure 8). Considering that the system is at design stage, the accuracy of speed measurements is still pending.

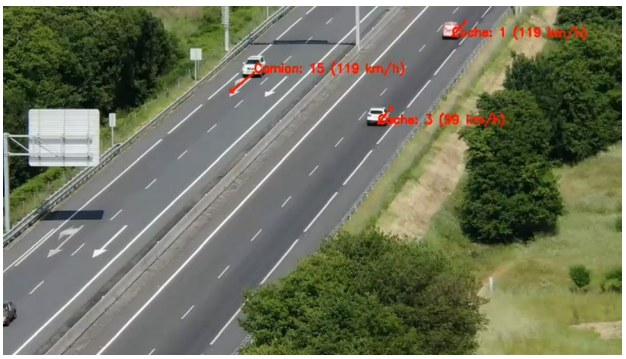

Figure 8 Estimation of vehicle speed from the reprojection of the image to ground coordinates

3.3 On-the-edge implementation results

The primary motivation behind implementing inference optimizers lie in achieving reduced latency and higher throughput for the classification. Latency refers to the time delay between inputting data into the model and receiving the corresponding output. By optimizing the inference process, these optimizers aim to minimize this delay, enabling near real-time or even real-time performance depending on time-sensitivity.

In addition to latency reduction, inference optimizers also strive to achieve higher throughput. Throughput, in this context, refers to the number of inferences or predictions that can be processed per unit of time by the inference engine and hardware-specific optimizations on GPUs.

Table 2 shows the improvements in processing time, including preprocessing of the images, the inference stage and postprocessing.

These tests were conducted on an NVIDIA GPU 3080 Laptop. The case study aimed to evaluate the impact of inference optimization on the overall performance of the model.

	mAP50-95 (Original)	mAP50-95 (TensorRT without quantification)	mAP50-95 (TensorRT with quantification)
Preprocessing	2.5 ms	2.6 ms	2.5 ms
Inference	23.3 ms	19.5 ms	10.4 ms
Postprocess	1.9 ms	2.1 ms	2.1 ms
Total	27.7 ms	24.2 ms	15.0 ms

Table 2 Processing time for on-the-edge implementation

The results demonstrated a significant improvement in inference throughput, with a notable x1.8 enhancement achieved. This improvement indicates a substantial reduction in the time required for making predictions, enabling faster and more efficient object detection.

Although there was a slight increase in preprocessing and postprocessing time due to memory copy bottlenecks between devices, the overall inference time was consistently reduced. This highlights the effectiveness of the inference optimization techniques implemented, which successfully mitigated any potential delays caused by data transfer or synchronization between different memory spaces.

Layer fusion and data quantization techniques, while beneficial for optimizing inference throughput, may introduce a potential trade-off in terms of precision. These optimization methods involve merging multiple layers into a single operation and reducing the precision of numerical data to achieve faster computations and memory usage. However, this reduction in precision can result in a loss of accuracy or subtle differences in the output. Table 3 shows the precision of the optimized results, showing negligible difference with the original.

	mAP50-95 (Original)	mAP50-95 (TensorRT without quantification)	mAP50-95 (TensorRT with quantification)
Car	0.3420	0.3420	0.3410
Motorcycle	0.0825	0.0821	0.0809
Truck	0.0230	0.0231	0.0230
All	0.1490	0.1490	0.1480

Table 3 mAP after on-the-edge optimization

4. CONCLUSIONS

In this work we presented the design monitoring system based on an UAS and an intelligent UAS-payload for on-the-edge processing and analysis of video frames to detect incoming traffic. The preliminary results for the tests show the potential of the solution. Particularly, we have shown that incorporating synthetic data from virtual reality frameworks into the training process significantly improved the performance and robustness of the machine learning models. We analysed and found that the optimizations for on-the-edge analysis of images led to a reduction in the processing time that permits near real-time operation. At the same time, the mean Average Precision of the results was not significantly stricken by the optimizer, showing metrics differences that are negligible. Finally, a coarse definition of the vehicles' position based on geometric simplifications and assumptions led to promising results on speed estimation, but further efforts and assessment are pending as future work for a more robust performance.

ACKNOWLEDGEMENTS

This research has received funding from the Government of Spain through project "A bottom-up digitalization approach to Green-Gray Transport Infrastructure Maintenance based on Deep Learning and Information Modelling", with a unique reference number TED2021-132000B-I00, from the component 17 of the Spanish economic recovery, transformation, and resilience plan - RETC funded by MCIN/AEI/10.13039/501100011033. This paper was carried out in the framework of the InfraROB project (Maintaining integrity, performance and safety of the road infrastructure through autonomous robotized solutions and modularization), which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 95533. It reflects only the authors' views. Neither the European Climate, Infrastructure, and Environment Executive Agency (CINEA) nor the European Commission is in any way responsible for any use that may be made of the information it contains.

REFERENCES

Conde, D., Martínez, J., Balado, J., & Arias, P., 2023. Generation of road zone synthetic data for training MOT models with the NVIDIA Omniverse platform. *The 30th EG-ICE: International Conference on Intelligent Computing in Engineering*, 722–730.

Dirección General de Carreteras., 2014. *Manual de ejemplos de señalización de obras fijas*.

Elistair | The Tethered Drone Company., 2023. Retrieved July 01, 2023, from https://elistair.com/

European Transport Safety Council, 2011. Preventing Road Accidents and Injuries for the Safety of Employees Work Related Road Safety Management Programmes.

Hussain, M., 2023. YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection. *Machines*, *11*(7), 677. doi.org/10.3390/machines11070677

Lin, Y., Wang, P., Ma, M., 2017. Intelligent Transportation System(ITS): Concept, Challenge and Opportunity, *IEEE International Conference on Intelligent Data and Security, IDS.* IEEE, pp. 167–172. doi.org/10.1109/BigDataSecurity.2017.50

Rao, A. S., Radanovic, M., Liu, Y., Hu, S., Fang, Y., Khoshelham, K., Palaniswami, M., & Ngo, T., 2022. Real-time monitoring of construction sites: Sensors, methods, and applications. *Automation in Construction*, Vol. 136. doi.org/10.1016/j.autcon.2021.104099

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A., 2016. You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR)*, 779–788. doi.org/10.1109/CVPR.2016.91

Rúa, E., Comesaña-Cebral, L., Arias, P., & Martínez-Sánchez, J., 2022. A top-down approach for a multi-scale identification of risk areas in infrastructures: particularization in a case study on road safety. *European Transport Research Review*, 14(1). doi.org/10.1186/s12544-022-00563-0

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., & Wang, X., 2022. Bytetrack: Multi-object tracking by associating every detection box. *European Conference on Computer Vision*, 1–21.