IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 April 2023, accepted 11 June 2023, date of publication 19 June 2023, date of current version 27 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3287770

RESEARCH ARTICLE

Automated Road Damage Detection Using UAV Images and Deep Learning Techniques

LUÍS AUGUSTO SILVA[1], (Member, IEEE),
VALDERI REIS QUIETINHO LEITHARDT[2,3], (Senior Member, IEEE),
VIVIAN FÉLIX LÓPEZ BATISTA[4], GABRIEL VILLARRUBIA GONZÁLEZ[1],
AND JUAN FRANCISCO DE PAZ SANTANA[1]

[1]Expert Systems and Applications Laboratory (ESALAB), Faculty of Science, University of Salamanca, 37008 Salamanca, Spain
[2]VALORIZA, Polytechnic Institute of Portalegre, 7300-555 Portalegre, Portugal
[3]Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, 1959-007 Lisbon, Portugal
[4]Department of Computer Science and Automatics, University of Salamanca, 37008 Salamanca, Spain

Corresponding author: Luís Augusto Silva (luisaugustos@usal.es)

The work of Luís Augusto Silva was supported by the Call for Predoctoral Contracts USAL 2021, co-financed by Banco Santander. This work was supported in part by the Spanish Agencia Estatal de Investigación, and in part by the Project Monitoring and Tracking Systems for the Improvement of Intelligent Mobility and Behavior Analysis (SiMoMIAC) under Grant PID2019-108883RB-C21/AEI/10.13039/501100011033, and supported by the Junta Castilla y León through the Project Investigación y desarrollo en tecnología y algoritmos inteligentes para plataforma unificada, distribuida y de bajo coste de monitorización de inmuebles e infraestructuras públicas o privadas a través de imágenes satelitales o de drones under Grant 07/18/SA/0022.

ABSTRACT This paper presents a novel automated road damage detection approach using Unmanned Aerial Vehicle (UAV) images and deep learning techniques. Maintaining road infrastructure is critical for ensuring a safe and sustainable transportation system. However, the manual collection of road damage data can be labor-intensive and unsafe for humans. Therefore, we propose using UAVs and Artificial Intelligence (AI) technologies to improve road damage detection's efficiency and accuracy significantly. Our proposed approach utilizes three algorithms, YOLOv4, YOLOv5, and YOLOv7, for object detection and localization in UAV images. We trained and tested these algorithms using a combination of the RDD2022 dataset from China and a Spanish road dataset. The experimental results demonstrate that our approach is efficient and achieves 59.9% mean average precision mAP@.5 for the YOLOv5 version, 65.70% mAP@.5 for a YOLOv5 model with a Transformer Prediction Head, and 73.20% mAP@.5 for the YOLOv7 version. These results demonstrate the potential of using UAVs and deep learning for automated road damage detection and pave the way for future research in this field.

INDEX TERMS UAV, road damage detection, deep learning, object-detection.

I. INTRODUCTION

Managing the maintenance of all the roads in a country is essential to its economic development. A periodic assessment of the condition of roads is necessary to ensure their longevity and safety. Traditionally, state or private agencies have carried out this process manually, who use vehicles equipped with various sensors to detect road damage. However, this method can be time-consuming, expensive, and dangerous for human operators. To address these challenges, researchers and engineers have turned to Unmanned Aerial Vehicles (UAVs) and Artificial Intelligence (AI) technologies to automate the process of road damage detection. In recent years, there has been a surge of interest in using UAVs and deep learning-based methods to develop efficient and cost-effective approaches for road damage detection.

Unmanned aerial vehicles have proven to be versatile in various applications, including urban inspections of objects and environments. They have been increasingly used for road inspections, offering several advantages over traditional methods. These vehicles are equipped with high-resolution cameras and other sensors that can capture images of the road surface from multiple angles and heights, providing a comprehensive view of the condition of the road. Additionally, UAVs can cover a large area relatively quickly, reducing the need for manual inspections, which can be dangerous

The associate editor coordinating the review of this manuscript and approving it for publication was Halil Ersin Soken.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

for human operators. As a result, the use of UAVs for road inspections has gained significant attention from researchers and engineers. Combining UAVs with artificial intelligence techniques, such as deep learning, can develop efficient and cost-effective approaches for road damage detection. It is frequently mentioned as being utilized for urban inspections of things like swimming pools [1], rooftops [2], vegetation [3], and urban environments [4], [5].

Currently, road condition inspections in Spain are performed manually, requiring personnel to travel along roads to identify damage points. This method incurs high costs due to the need for human labor and specific cameras and sensors for the task. The decision-making process for repairing road damages is the responsibility of an expert. In contrast, countries like China have a vast network of roads and highways, making them susceptible to surface cracks and rainwater infiltration, which can accelerate the deterioration of roads and pose risks to vehicle safety. Without timely detection and the rapid availability of information on road defects, excessive wear on vehicles and an increased likelihood of traffic accidents can occur, leading to further financial losses. Therefore, the development of automated techniques for detecting road deterioration has become a critical area of research, with many universities and research centers collaborating to find effective solutions.

Automatic road damage detection is an active area of research that aims to detect and map various types of road damage using multiple techniques such as vibration sensors, Light Detection And Ranging (LiDAR) sensors [6], and image-based methods. These techniques are often used in combination to improve the accuracy of damage detection. Machine learning approaches, such as deep learning, are commonly used in image-based techniques to recognize various types of road degradation. These methods typically require a dataset of images, which can include top-down photographs, images captured by unmanned aerial vehicles [7], pictures obtained by mobile devices [8], [9], images obtained from satellite image platforms [10], thermal images [11], and 3D images or stereo vision of the asphalt surface [12].

Researchers have been conducting studies using a variety of datasets to train the model, incorporating additional images captured by drones, cameras mounted on cars, and satellites. To facilitate the learning process, these datasets are often annotated to identify different types of road damage, including, but not limited to potholes, cracks, and rutting. Annotating these images enables the algorithm to learn to detect and classify various types of road damage accurately. Using a large and diverse dataset, researchers can enhance the accuracy and reliability of their models, ensuring that they can effectively identify and address different types of damage on the roads.

A. THE ROAD DAMAGE DETECTION DATASET

To support the development of automated road damage detection techniques, the Crowdsensing-based Road Damage Detection Challenge (CRDDC) [13] was organized as part of the IEEE BigData Cup 2022. This international competition involves a published dataset of 47,420 road images from six countries: Japan, India, the Czech Republic, Norway, the United States, and China. The images have been annotated with more than 55,000 instances of road damage, including longitudinal cracks, transverse cracks, alligator cracks, and potholes.

CRDDC aims to encourage the development of deep learning-based methods to detect and classify road damage automatically. Municipalities and road agencies can utilize the RDD2022 dataset for low-cost automatic monitoring of road conditions. Moreover, computer vision and machine learning researchers can use the dataset to benchmark the performance of different algorithms for other image-based applications of the same type, such as classification and object detection.

Several organizations used the RDD2022 dataset for their models, while some excluded the China Drone portion of the dataset. The top algorithms used by these organizations include YOLO-series and Faster RCNN-series models, YOLOv5, YOLOv7, and YPLNet.

Many organizations used ensemble models to achieve better accuracy, with techniques such as image patch strategies, customized anchor boxes, attention modules, and ensembling models trained with multiple levels of augmentations. Other techniques include image augmentations, label smoothing, coordinate attentions, cropping Norway images to focus only on road areas, and training country-specific models using data from all countries.

B. THE YOLO-SERIES

The evidence in the literature presents You Only Look Once (YOLO) as one of the most used algorithms in the object detection field. It is a popular object detection algorithm, and several versions have been released. When we compare the evolution of all the YOLO series, we can see a significant evolution concerning detection time. In the first version published [14] Since just a single back-propagation neural network is required to make a prediction, the YOLO is made to run on devices with low processing power. Since the initial version was based on AlexNET, this method has undergone several more iterations.

In the timeline of the YOLO algorithm, YOLOv3 [15] and the YOLOv4 version [16] appear. In summary, YOLOv3 and YOLOv4 are both deep learning-based object detection algorithms, but YOLOv4 is an improvement over YOLOv3. YOLOv4 has been optimized for real-time object detection and trained on a large dataset of images and videos to improve its accuracy. YOLOv4 also includes new techniques, such as Mosaic data augmentation and DropBlock, enhancing its performance.

The YOLOv4 is considered the latest and most accurate version of YOLO till 2021. It is built on a custom-designed neural network architecture that uses a combination of convolutional and transposed convolutional layers to detect objects in images and videos. YOLOv4 has been optimized for real-time object detection and trained on a large dataset of images and videos to improve its accuracy.

Subsequently, the fifth version of the algorithm, called YOLOv5 [17], was released. This algorithm turned out to be a perfect model, bringing more options as we can highlight the image segmentation, but it still needs to be closer to the 5th major update. The results are very similar to YOLOv4 which a considerable amount of work was done, and all the nuances were taken into account. YOLOv5 is an improvement over YOLOv4. It is based on a new SPADE architecture, which uses semantic and spatial information to improve object detection accuracy. YOLOv5 also uses a new training algorithm called Mosaic Data Augmentation to enhance the model's generalization.

Later and more recently, the seventh version of the algorithm was released [18], the latest iteration in the life cycle of YOLO models. YOLOv7 infers faster and more accurately than its previous versions (i.e., YOLOv5). YOLOv7 is the latest version of YOLO. It has been built on a new architecture called Efficient-YOLO, which uses EfficientNet as the backbone network. YOLOv7 has been trained on a large dataset, and it has been optimized for real-time object detection. It is more accurate and faster than previous versions of YOLO.

In conclusion, YOLOv4 is considered the most accurate version of YOLO until 2021 and is optimized for real-time object detection. YOLOv7 is the latest version of YOLO, and it is based on a new architecture called Efficient-YOLO, which is more accurate and faster than previous versions.

C. OBJECTIVES AND STRUCTURE

This paper builds upon a previous project proposing an architecture for a pavement monitoring system with pothole recognition in UAV images [7]. In this new research, we expand upon the previous solution by comparing it with new algorithms and datasets, introducing new classes of damage, and adopting data augmentation during training, which promotes adapting to dramatic size changes of objects in images. Finally, in this work, the YOLOv5 and YOLOv7 are compared, and an improvement was made in the YOLOv5 model using the Transformer Prediction Head for the UAV use case.

We have used a merged dataset from previous work and Crowdsensing-based Road Damage in this work. Detection Challenge, including new damage classes for a more comprehensive understanding of pavement damage. Experimental results demonstrate the effectiveness and efficiency of our proposed solution, achieving more accuracy on the test dataset.

The main objective of this project is to improve the autonomous monitoring system for the state of roads using images captured by drones and advanced artificial vision and intelligence techniques. The proposed system will notify the maintenance company about detected road damage, including the ability to send messages with the geographical coordinates of the damages found.

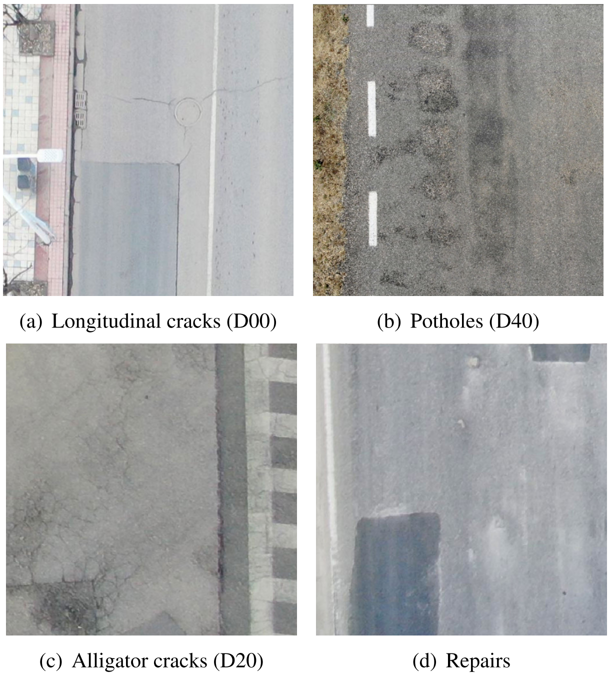

(a) Longitudinal cracks (D00)
(b) Potholes (D40)
(c) Alligator cracks (D20)
(d) Repairs

FIGURE 1. Road damage classes in the dataset.

Our team has made several contributions, including:

- Adding an extra prediction head to address the issue of large variations in object scales.
- Incorporating Transformer Prediction Heads (TPH) into the YOLOv5 model, resulting in improved object localization in high-density scenes.
- Providing a range of useful techniques and filtering out ineffective approaches for object detection in drone-captured scenarios.
- Enhancing the classification accuracy of certain ambiguous categories by utilizing a self-trained classifier.
- The project has introduced several new classes of pavement damage, as depicted in Figure 1. These include longitudinal cracks, alligator cracks, potholes, bumps, and repairs. The project offers a more comprehensive understanding of pavement damage by including these additional classes, enabling more precise and efficient road infrastructure monitoring.

In the overall project, convolutional neural networks detect asphalt defects, allowing for operator overrides or suggestions for improved accuracy over time. Additionally, we will implement a feature for automatically planning routes to cover the entire road, eliminating the need for manual operation by the pilot and utilizing PIX4D to automate the route planning.

The structure of this paper is as follows: Section II thoroughly analyzes existing literature on damage detection methods and UAVs. Section III delves into the proposed system's architectural design, the dataset used, and its implementation. The experiments carried out, and their results are discussed in Section IV. Finally, Section V concludes the paper by presenting a summary of the findings and outlining potential future work.

II. RELATED WORKS

Imagery capture plays a crucial role in the initial assessment of a road or highway's condition. A UAV, specifically a drone, is an efficient and cost-effective way to capture high-quality and detailed photographs of the road surface from various perspectives. In this study, we have used the DJI Mavic Air 2S drone, a more recent drone version budgeted for this project. This drone has advanced features such as a high-resolution camera, GPS, and obstacle avoidance sensors, enabling it to capture high-quality road surface images with minimal distortion. Additionally, using a UAV allows for more comprehensive coverage of the road surface, especially in hard-to-reach areas, and can be done safely and quickly.

Related articles focus on improving existing algorithms in deep learning and unmanned aerial vehicles (UAVs). For example, autonomous UAVs have been used for structural health monitoring and real-time damage mapping using deep learning methods and ultrasonic beacons with geo-tagging [19], [20]. Deep learning techniques, such as CNNs, have shown promising results in various domains, including vehicle traffic monitoring [21], large population monitoring [22], animal identification [23], wind generator inspection [24], and electric component detection [25]. These techniques can also be used to analyze images or video from cameras mounted on vehicles to detect road potholes, making them an effective approach for automated road damage detection.

The transportation industry is no exception, and the task of road damage identification is ready to profit from the rapid advancement and diffusion of deep learning technologies. Using convolutional neural networks (CNNs) or other deep learning techniques to analyze images or video from cameras mounted on vehicles to detect road potholes is possible. One of the fundamental approaches for automated road damage detection is using deep learning algorithms. These algorithms effectively detect a range of objects, including damage.

Standard deep learning methods in this area include the implementation of Convolutional Neural Networks (CNNs). In the paper [26], the authors proposed a deep convolutional neural network (CNN) for road damage detection from UAV images. The proposed CNN was trained and tested on a dataset of UAV images, and the results showed that it could detect road damage accurately. In [27] proposes a novel approach for detecting concrete cracks using a deep architecture of CNN without the need for image processing techniques (IPTs) to extract defect features. The CNN is trained on a large dataset of 40.000 images and achieves an accuracy of about 98%. The proposed method is tested on a different structure under various conditions and performs better than traditional Canny and Sobel edge detection methods.

In another recent work [28], the authors proposed a deep learning-based object detection method for automated road damage detection using UAV images. They used the Faster R-CNN algorithm as the object detector. Results reflected that the proposed method is superior to other methods of road damage detection.

Also, with Regions with Convolutional Neural Network (R-CNN) and their improvements called Faster R-CNN, the authors in [29] and [30] proposes for structural visual inspection, which can detect multiple types of damages, including concrete cracks, steel corrosion, bolt corrosion, and steel delamination. The proposed method achieves an average precision rating of 87.8%. The proposed method provides a remarkably fast test speed of 0.03 seconds per image and can potentially be used for quasi-real-time damage detection on video using the trained networks.

Finally, [31] proposes a crack detection and quantification method using Faster R-CNN and modified TuFF and DTM algorithms. The proposed method achieved high accuracy with 95% average precision, 83% intersection over union, and 93% accuracy for crack length.

In [32], the authors developed a new sensor technology for road damage detection using a deep learning-based image processing algorithm with super-resolution and semi-supervised learning methods based on GAN. Tested on 400 road images, the proposed method showed an average recognition performance of 81.54% and 79.228% in terms of mean intersection over union and F1-score, respectively. The paper suggests that the proposed method can be used for efficient road management in the future.

Nowadays, simply detecting damage in structural images is not enough. To fully understand and assess the extent of the damage, it is necessary to quantify it by measuring the size of the detected defects. This requires a more advanced technique known as pixel-level segmentation, which can accurately delineate the boundaries of the damaged areas in the image.

Kang [33] proposes a novel semantic transformer representation network (STRNet) for crack segmentation in complex scenes, achieving high performance and fast processing speed. The network was evaluated and compared with other advanced networks, showing superior performance and processing speed compared with other networks, including attention on [34], which proposes a high-performance deep-learning network for real-time pixel-level segmentation of internal damages in concrete members using active thermography. The attention-based IDSNet outperforms state-of-the-art networks with a mean intersection over the union of 0.900, a positive predictive value of 0.952, an F1-score of 0.941, and a sensitivity of 0.942.

Single-Shot Detection (SSD) is another point-of-view specifically for road or concrete damage detection. The work [35] presents the SDDNet, a deep learning model for real-time segmentation of concrete cracks in images, achieving high accuracy on a manually created dataset. The model is compared with recent models and outperforms them while processing images at 36 FPS, which is significantly faster than previous works.

In another work, Arya et al. [36] reported a set of state-of-the-art solutions in global road damage detection and classification tasks. For example, Pham et al. [37] experimented with a study with Detectron2, implementing Faster R-CNN. Generally, these reviewed studies show that the Faster R-CNN model provides better accuracy with the trade-off of prediction time (8 frames per second) than the YOLO model (40 frames per second). In contrast, SSD balances the two concerning prediction accuracy and time.

A. YOLO IMPLEMENTATIONS

In [38], presents a deep learning approach to identifying potholes on Indian roads using the YOLO algorithm to improve road maintenance and reduce accidents. A new dataset of 1500 images of Indian roads is created and trained using YOLOv3, YOLOv2, and YOLOv3-tiny, and the results are compared in terms of accuracy. In contrast, the authors in [39] present the M-YOLO, which uses a light network architecture based on MobileNet V3 and YOLOv5S to improve the detection efficiency of pavement oil repair using UAV images. The results of experiments showed that the M-YOLO algorithm has an accuracy of 98.3%, an average accuracy of 95.5%, and a detection speed of 96.6fps, which is significantly better than YOLOv3 in terms of accuracy, speed, and number of parameters.

In addition, the authors in [12] present a novel automated pavement distress detection framework that combines stereo vision and deep learning. The proposed method is tested on asphalt roads under various conditions. The results show that it can achieve millimeter-level accuracy in the crack and pothole segmentation and that the enhanced 3D crack segmentation model is superior to other models in terms of accuracy and inference speed. It also uses the high-resolution pothole segmentation map to measure the pothole volume accurately.

Another solution related to the literature is using multispectral images to detect road damages [40]. Multispectral imaging using UAVs is a powerful tool for detecting and analyzing road damage. Another approach is using hyperspectral images to detect road pavement cracks. In the study [41], an asphalt crack index is introduced and found to be effective for crack detection, with an average 21.37% increase in F1-score compared to the existing metric in literature.

Convolutional Neural Networks (CNNs) and Transformer can also be used for hyperspectral image classification [42]. CNNs can extract features from hyperspectral data by learning spatial patterns in the spectral domain, while Transformers can capture global contextual information by modeling long-range dependencies. Both approaches have shown promising results in hyperspectral image classification tasks.

In the literature, many works are connected to this field, which is developing very fast, and more experiments should be performed to find a better approach in this specific case. In this work, we approached YOLO because it is the most efficient technique. When we wrote the first article, the most recent version was YOLOv4. YOLOv7 is the current version of this algorithm, which has been tried extensively in this work. Currently, the work of Wang et al. [18] is the official implementation, which some of the authors are the same as the YOLOv4 version [16].

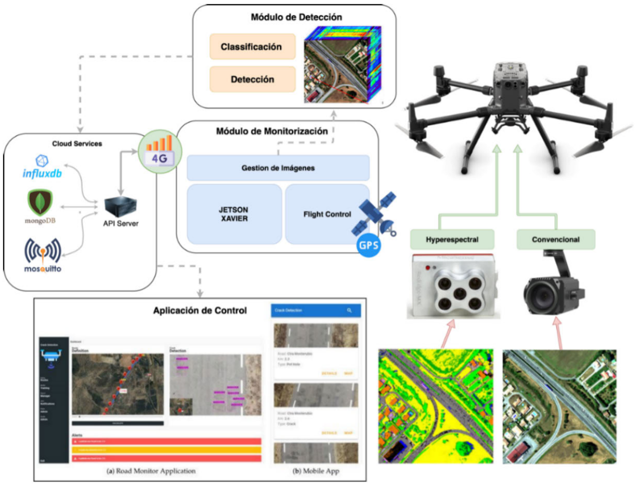


FIGURE 2. Design of the proposed method.

The speed and precision of YOLOv7 are between five and 160 frames per second. This project tested various hyperparameters (using the freebies) and models (adding modules and custom configuration files) to train models for road damage detection and classification tasks.

III. DESIGN AND IMPLEMENTATION

A. IMPLEMENTATION

The proposal's main objective is the detention of deformities on street surfaces, roads, highways, and other vehicle traffic surfaces. The initial proposal of this project, as seen in Figure 2, uses a commercial drone integrated with a high-resolution camera and, in its case, also the use of a multispectral camera. The multispectral camera, as its name suggests, is a camera that is capable of capturing several light spectra. In the case of the dataset of this article, the use of a multispectral camera is not involved and uses only images from high-resolution cameras.

B. UAV IMAGE DATASET

First, we conducted a literature search to find a dataset of potholes and cracks in asphalt at the outset of this study. However, the databases were different from the current suggestion of utilizing an unmanned aerial aircraft to take photographs at a safe distance from the road. Therefore, a new dataset was required to depict the Spanish road situation accurately. In total, 600 pictures with a resolution of 3840 × 2160 pixels were taken. The images were taken from a DJI Air 2S drone 50 meters from the ground on roads in Spain and only had two classes, potholes (D40) and cracks (D00).

Upon dataset creation and labeling of all photographs, 568 tagged photos were recovered. The photos' orientation was adjusted during the pre-processing stage and got a new size (640 × 640). Different iterations of each image in the collection were created utilizing augmentation techniques.

TABLE 1. Spain roads dataset.

Class	Spain Annotations
D00 (Crack)	327
D40 (Pothole)	3480

The zoom levels of the photographs ranged from 0% to 15%. In total, 1362 images are included in the collection. 70% of these photos were used for training, 20% for validation, and 10% for testing the trained model's efficacy. This dataset was used in previous work [7], and its repository is available.[1] Table 1 shows how the classification is organized.

Continuing to compose the dataset, we approached the previous datasets (Spain) as a reference for training deep learning models to detect road damage from the collected videos automatically. Added to this, we joined the dataset provided in the CRDDC2022. This dataset is a dataset of road damage in multiple countries [43].

This benchmark dataset is used for training and testing machine learning models for automated pavement distress detection. The dataset contains 47,420 road photos from five countries (China, Japan, the Czech Republic, Norway, the US, and India). We use these photos to train and test models to identify four types of pavement damage: alligator cracks (D20), transverse cracks (D10), longitudinal cracks (D00), and pothole cracks (D40).

The training set of this dataset is used to train machine learning models to recognize the four types of pavement damage. The models learn to identify the characteristics of each type of damage from the photos in the training set. The testing set of this dataset is used to evaluate the performance of the trained models. The models are applied to the photos in the testing set, and their predictions are compared to the actual labels to evaluate the model's accuracy.

This dataset is useful for researchers and engineers working on automated pavement distress detection because it provides a large and diverse set of images that can be used to train and test models. Including images from different countries ensures that the models trained using this dataset can generalize well to different road conditions and environments.

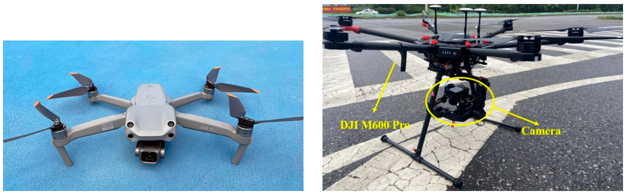
These images were obtained from smartphones, high-resolution cameras, and satellite images. All are obtained by employing cars, motorcycles, and drones. The distribution of damage types (of the four relevant damage types) by countries is displayed explicitly in Table 2. For China, two datasets were made available: Ch_M, which refers to images taken by mobile phones, and Ch_UAV, which refers to images taken by drones.

To compose the dataset for this article, we used the first dataset of roads in Spain and a small part of the images taken by drones in China **Ch_UAV**, as mentioned in the table above. This dataset also includes two complementary classes **Repair** that refers to some repair done on the road and **Block Crack**.

[1] https://github.com/luisaugustos/Pothole-Recognition

TABLE 2. Damage category-based data statistics for RDD2022.

	JPN	India	CZ	NW	US	Ch_M	Ch_UAV
D00	4049	1555	988	8570	6750	2678	**1426**
D10	3979	68	399	1730	3295	1096	**1263**
D20	6199	2021	161	468	834	641	**293**
D40	2243	3187	197	461	135	235	**86**



(a) Air 2S is DJI used to adquiry images in Spain.

(b) The drone and camera set-up used to capture China_Drone data included in RDD2022.

FIGURE 3. UAV used to obtain images for the dataset.

TABLE 3. Damage category-based data statistics for the merged dataset.

Class	China_Drone	Spain	Total
D00	1426	327	1753
D10	1263	0	1263
D20	293	0	293
D40	86	3480	3566
Repair	769	0	769
Block Crack	3	0	3

TABLE 4. Dataset split.

	Number of Images	Percentage
Training Set	4000	82%
Validation Set	584	12%
Testing Set	289	6%
TOTAL	4873	100%

We noticed that including the China_Drone data in the proposed training set increased the dataset's heterogeneity. They were aligned with RDD2020 and this work, which focuses on low-cost and affordable automatic road damage detection considering feasible methods for the public.

The final dataset has 2893 images and comprises images from both countries obtained by UAV detailed in Fig. 3(a) and Fig. 3(b).

Table 3 provides an overview of the damage category-based data statistics for the merged dataset, including the distribution of classes and annotations for the China_Drone and Spain datasets.

This dataset is augmented and preprocessed using auto-orientation and also resized to 640 × 640. The augmentation was performed on the images to increase the size and diversity of the dataset artificially. Each training example has been augmented to produce two outputs in this case. The rotation has been applied to randomly rotate the images between −15° and +15° to make the model more robust to different orientations of the objects being detected. The final dataset description was presented in Table 4.

C. DATA PREPARATION

According to the previous table, the two data sets were combined and divided into three versions formatted for YOLOv4, YOLOv5, and YOLOv7. One directory is needed for training, while the other is for validation. Additionally, these two folders must have the label and image directories. The labels would include a text file holding the image annotation for each labeled image, while the images would contain the actual photos. The text file's name must match that of the associated image. After producing new YOLO annotations, the folder follows the YOLO dataset structure. Information about the data set, including names and the number of classes, is contained in the file "data.yaml". All this was done thanks to the platform Roboflow,[2] on which all datasets are stored.

D. MODEL TRAINING

The YOLOv4-tiny model served as the initial basis for this work, which adheres to the coordinated prediction concept just like YOLOv2 and YOLOv3 did. Multi-class classification is possible instead of single-class classification, as in the older versions. This initial network was set up to detect two classes within the 568 images. Later with the use of YOLOv5, YOLOv5-Tranformer, and YOLOv7, the number of classes increased to 6. Afterward, new training was performed with the six classes and the 4000 images.

To train our YOLO models, we prepared our data and fed them with the necessary data set. The model trained is capable of detecting the following sorts of cracks: longitudinal cracks (D00), transverse cracks (D10), alligator cracks (D20), pothole cracks (D40), and repair and block cracks. We used 4873 photos from the dataset generated by the roboflow platform to train the model. The training and validation of the models described in this research were carried out using an Intel(R) Core(TM) i9-10940X CPU @ 3.30GHz computer with 128GB of RAM and an RTX3090 GPU with 24GB of integrated memory due to the availability of reasonably priced GPUs.

E. IMAGE AUGMENTATIONS

Image augmentation is a technique to expand the training dataset by applying various transformations to the existing images. Image augmentation aims to introduce variability and diversity in the training dataset, which helps improve the model's generalization ability. YOLOv7 and YOLOv5 can use various image augmentation techniques, such as:

- **Random horizontal flipping:** This technique randomly flips the image horizontally, giving the model more examples of the same object in different orientations.
- **Random cropping**: This technique randomly crops a portion of the image, giving the model more examples of the object in different scales and positions.
- **Random rotation**: This technique randomly rotates the image, providing the model with more examples of objects in different orientations.

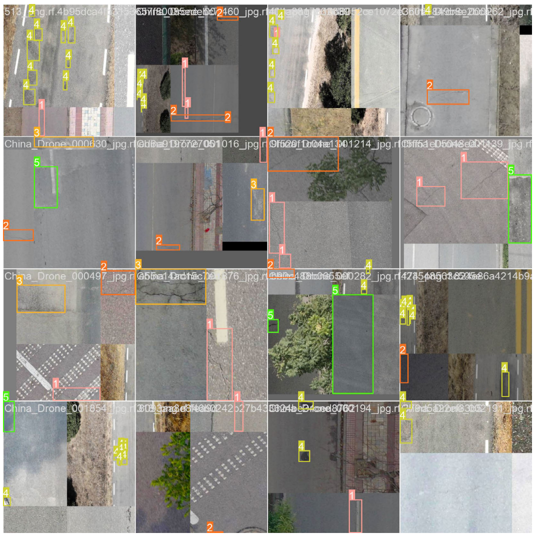


FIGURE 4. YOLOv5's default parameters on augmented images, showcasing the potential for image distortion and inconsistencies.

- **Random brightness and contrast**: This technique randomly adjusts the brightness and contrast of the image, providing the model with more examples of the object under different lighting conditions.
- **Random color jitter**: This technique randomly changes the image's color, giving the model more examples of the object under different color variations.

YOLOv5 and YOLOv7 are state-of-the-art algorithms that enhance and upscale images to improve the robustness and accuracy of the model. However, as seen in Figures 4 and 5, several issues with the default parameters negatively impact the results. Using these techniques, YOLOv7 and YOLOv5 can increase the size and diversity of the training dataset. This can help prevent overfitting and improve the model's generalization ability.

Three models were trained: the first using a YOLOv4 architecture, which replicated the experiment from the prior study; the second with a YOLOv5 design; and the third with a YOLOv7 architecture. We will now go over each model's outcomes and guess how and why we came to those conclusions to choose the ultimate design that is most appropriate for the task at hand.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

Intending to test our system, we compare the labeled dataset photos with the final images identified by the algorithm, paying attention to quantitative characteristics. In this case, different experiments result in different models, and in the case of using three different models, we must pay attention to the evaluation process.

Therefore, we need a robust metric to select the best models among all experiments. There are two standard evaluation metrics used in this area. The first is the Mean Accuracy (mAP) calculated in IoU (Intersection overlapping) limit of 0.5 (mAP@0.5). The second is the F1 score. The MAP is

[2] https://roboflow.ai

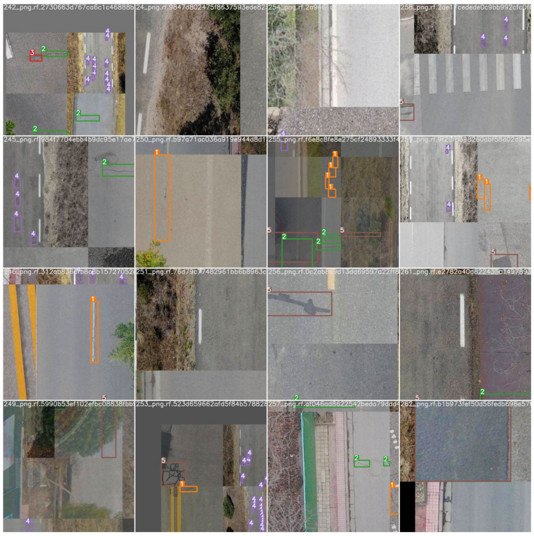


FIGURE 5. YOLOv7's default parameters on augmented images, showcasing the potential for image distortion and inconsistencies.

a good measure when we must ensure the model is stable at different confidence limits (robust) while the F1 score is computed for a specific confidence limit. The common practice is to use mAP@0.5 in the validation set to select the best model and use the F1 score to report the model performance in the test data set. This project also follows this common practice (using mAP@0.5 to select the best models and report the F1 scores in the test sets).

In terms of the quantitative assessment, we'll employ the following metrics: the precision, or the ratio of true positives (TP) to true positives (TP) plus false positives (FP) Equation 1. Equation 2 combines the recall, the likelihood that a picture will be categorized as positive, and the ratio of true positives (TP) to true positives plus false negatives (FN). The third and final metric is the F1 metric, which combines the first two abovementioned metrics.

On the other hand, we also have the classification speed, which is expressed in frames per second (FPS), the mean average precision (mAP), which is determined by using the precision and recall curve, and the IoU, or the overlap area between the detected and imaged areas.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{3}$$

The model training process was evaluated in 3 steps, switching between iterations and image resolution. The measures mAP@0.5 are used to select the best models during training based on the validation data.

Additionally, with the main evaluation process, we introduced four ways to evaluate the comparative analysis and the implementation. Using hyperparameter tuning (evolve), error analysis, transfer learning, and ensemble methods.

TABLE 5. Performance metric for YOLOv4.

Label	TP	FP	AP (%)
Block crack	0	0	0.00%
D00	87	106	27.14%
D10	149	232	35.17%
D20	4	14	10.39%
D40	162	42	44.25%
Repair	63	69	44.24%

- Hyperparameter tuning: This involves adjusting the settings and parameters of a model or algorithm to find the best configuration for a given task. By systematically varying the values of different parameters, researchers can determine which settings lead to the best performance and gain insight into which factors are most important.
- Error analysis: This involves examining the errors made by a model or algorithm and identifying patterns or trends. By analyzing the specific cases where a system fails, researchers can better understand its limitations and identify areas for improvement.
- Transfer learning: This involves using a pre-trained model or algorithm as a starting point and fine-tuning it for a specific task. By leveraging the knowledge and experience encoded in a pre-trained model, researchers can often achieve better results with less data and training time.
- Ensemble methods: This involves combining the predictions of multiple models or algorithms to improve overall performance. By leveraging the strengths of different models and compensating for their weaknesses, ensemble methods can often achieve better results than any individual model.

A. YOLOv4 EXPERIMENTS

With YOLOv4, the first processing step was performed. Convolutional layers were adjusted as necessary using a pre-trained weight model. Compared to the earlier work, the results of this training could have been better, implying that the prior work involved overtraining or overfitting. In this instance, we achieved a precision of 0.50, with a recall of 0.32 and an F1 score of 0.39. They chose a mean average precision (mAP@0.50) of 0.268638, or 26.86%, for this training, in just three seconds of detecting time. The performance is broken down per class in Table 5.

B. YOLOv5 EXPERIMENTS

Employing YOLOv5 v7.0 for road damage detection showed significant improvements compared to YOLOv4, as indicated in Table 6. The mean average precision (mAP) at an IoU threshold of 0.5 (mAP@.5) increased from 26.86% to 59.90%, indicating a substantial increase in the model YOLOv5x ability to detect road damage accurately.

TABLE 6. Performance metric for YOLOv5.

Class	Imgs	Labels	P	R	mAP@.5	mAP@.95
all	584	1447	0.787	0.561	0.599	0.344
Block	584	1	1	0	0.0203	0.0142
D00	584	367	0.68	0.557	0.594	0.294
D10	584	248	0.778	0.742	0.829	0.46
D20	584	65	0.702	0.507	0.566	0.299
D40	584	623	0.814	0.785	0.817	0.419
Repair	584	143	0.745	0.776	0.766	0.578

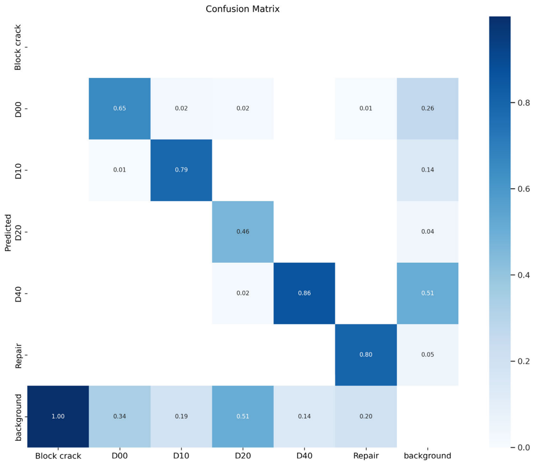


FIGURE 6. Confusion matrix for the YOLOv5 model.

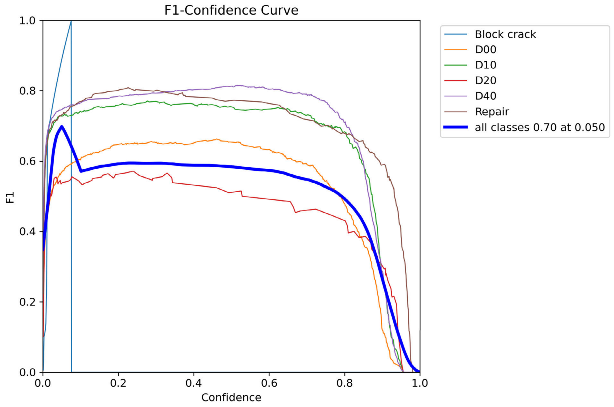


FIGURE 7. F1-Confidence curve.

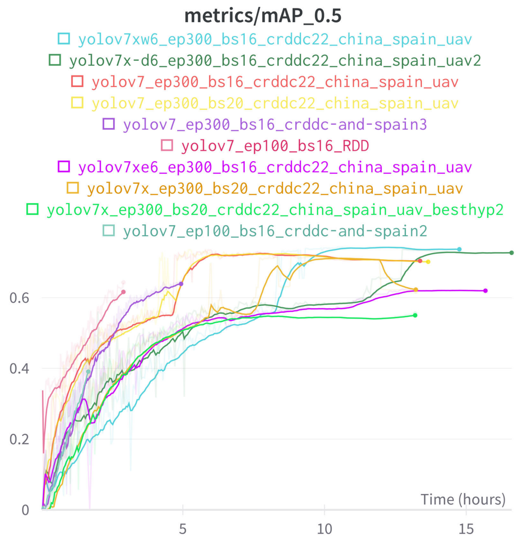


FIGURE 8. mAP@0.5 with the YOLOv7 versions implementing hyperparameters finetuning.

Additionally, the mAP at an IoU threshold of 0.5 and a recall threshold of 0.95 (mAP@.5:.95) also showed a significant improvement, with a boost of around 27%.

Furthermore, the recall percentage, which measures the proportion of actual positive instances that are correctly detected, also increased from 32% to 56.10%. These results demonstrate the effectiveness of YOLOv5 in detecting road damage with high accuracy and recall. Additionally, the inference time of YOLOv5 is 17.2 milliseconds, with a pre-processing time of 0.9 milliseconds, an inference time of 17.2 milliseconds, and a Non-maximal Suppression (NMS) of 6.8 milliseconds per image at shape (1, 3, 640, 640). These results show that YOLOv5 is accurate and efficient in processing time. As shown in Figure 6, the confusion matrix for classifying six groups using test data and YOLOv5 revealed that the model correctly classified most classes.

The horizontal axis represents the ground truth, and the vertical axis represents the predicted classes. The diagonal elements, which represent the correctly classified classes, are the highest among all elements in the matrix, indicating a high level of accuracy. However, it can also be observed that classes D10 and D40 have some misclassifications. These classes are overrepresented in the dataset, which could have made the model more sensitive to detecting these specific classes. However, overall, the results demonstrate the effectiveness of YOLOv5.

Additionally, we calculated the F1 score for each class, as shown in Figure 7. Compared to accuracy, this is a more effective metric. The six groups differ in look and structure, making the issue more challenging. The system successfully got a 70% total F1 score. However, the percentage of the F1 score allocated to each class varies from 50% to 80%. The D20 class has the lowest F1 score due to weaker recall, whereas the D40 class has a higher score. A conclusion that the earlier-presented Confusion Matrix supports.

C. YOLOv7 EXPERIMENTS

The best experiment using the YOLOv7 model was trained using the YOLOv7x-W6 model with 300 epochs. A batch size of 16 concluded after the grid research proving the accuracy of prediction and classification try the other YOLO models, YOLOv7-W6, YOLOv7-E6, YOLOv7-D6, YOLOv7-E6E. Figure 8 shows the ten performed trains with these models, changing the batch size and hyperparameters.

Overall, YOLOv7 achieved an mAP of 0.737, indicating that it performed reasonably well in detecting the different

TABLE 7. **YOLOv7 performance on pavement damage dataset.**

Class	Img	Labels	P	R	mAP@.5	mAP@.95
all	584	1463	0.788	0.714	0.737	0.453
D00	584	349	0.735	0.573	0.590	0.259
D10	584	247	0.751	0.708	0.731	0.348
D20	584	57	0.633	0.439	0.475	0.203
D40	584	627	0.814	0.769	0.802	0.395
Repair	584	182	0.797	0.799	0.827	0.615

classes of pavement damage. The highest mAP was obtained for the **Repair** class, which had an mAP of 0.827. The **Block crack** class achieved perfect precision and recall, indicating that the model could detect this type of damage accurately, but with the low presence of this class in all the datasets, this class was excluded.

However, some classes, such as **D20**, had lower precision and recall values, suggesting that the model struggled to detect this damage accurately. Nonetheless, the model achieved an overall precision of 0.788 and recall of 0.714, indicating that it could detect pavement damage in general accurately. The YOLOv7 is speedier, with an inference time of only 11.4 ms, demonstrating all of this. Speed per 640 × 640 image: 11.4/6.7/18.1 ms inference/NMS/total at batch-size 1

Table 7 shows the Precision (P), Recall (R), and mean Average Precision (mAP) scores for different classes of pavement damage.

In the Table, the detection effect of D00 and D20 is lower than that of other classes. This could be because both D00 (potholes) and D20 (alligator cracks) share some similarities in their visual appearance, which makes it more difficult for the model to distinguish between them and other classes.

Some specific strategies could be employed to improve the detection performance of YOLOv7 on D00 and D20. For example, data augmentation techniques could be used to increase the diversity of the training data, including variations in lighting, angle, and distance. Additionally, transfer learning could be applied to pre-train the model on a large dataset of similar objects, such as road surface images, before fine-tuning the specific classes.

Furthermore, the model architecture was adjusted in the YOLOv5 with Transformer Head to better handle the complexities and variations of potholes and alligator cracks. This improvement will be explained in the next subsections. As shown in Figure 9, the confusion matrix for classifying six groups using test data and YOLOv7 shows that the model accurately categorizes most classes. The horizontal axis represents the ground truth, and the vertical axis represents the predicted classes. The diagonal elements, which represent the correctly classified classes, are the highest among all elements in the matrix, indicating a high level of accuracy.

Compared to the YOLOv5 matrix, it can be observed that there is an increase in the D10 and D40 classes, which were overrepresented in the dataset. This could be due to the improved capability of YOLOv7 to detect these specific classes, even though they are overrepresented, which leads to improved overall identification. Overall, the results demonstrate the effectiveness of YOLOv7.

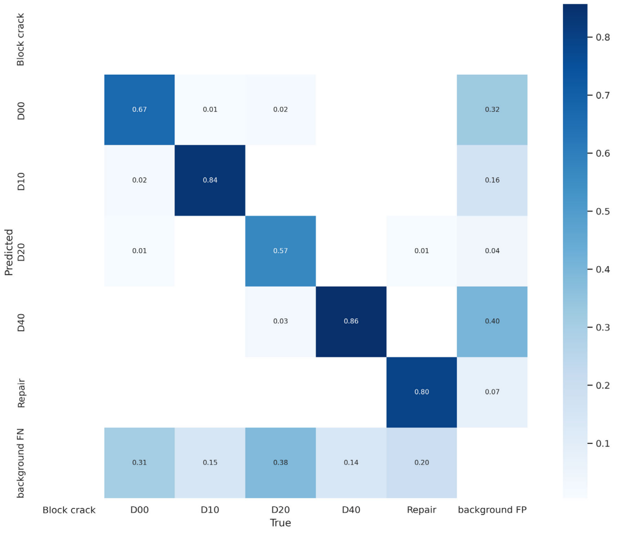


FIGURE 9. **Confusion matrix for the YOLOv7 model.**

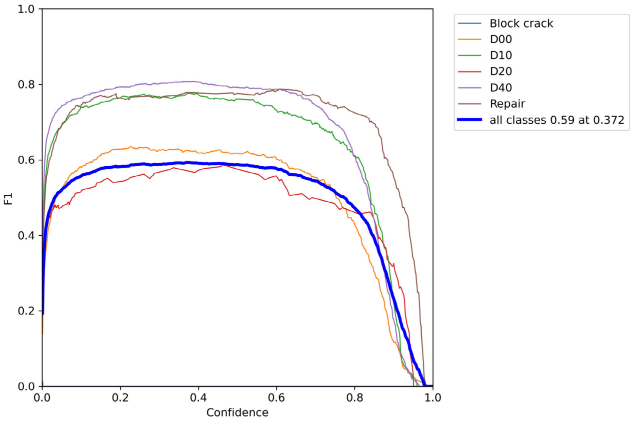


FIGURE 10. **F1-Confidence curve.**

The performance of the YOLOv7 algorithm is further analyzed by calculating the F1 score for each class, as shown in Figure 10. The system achieved a total F1 score of 59%. The individual classes have F1 scores ranging from 50% to 80%. The D10, D40, and Repair classes have higher scores than the others. Interestingly, the D00 class has a lower F1 score in YOLOv7 than in YOLOv5, as highlighted by the Confusion Matrix previously presented.

D. YOLOv5 + TRANSFORMER PREDICTION HEAD EXPERIMENTS

In our YOLOv5 + Transformer Prediction Head experiments, the transformer architecture has been used as the prediction head of the YOLOv5 model, replacing the conventional CNN-based prediction head. The transformer head takes in the features extracted by the YOLOv5 backbone network and generates predictions for object locations and class probabili-

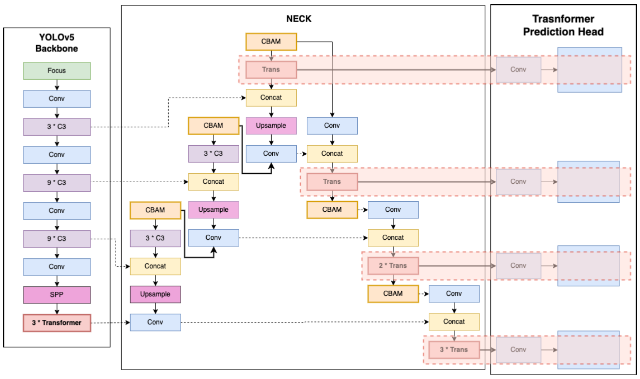


FIGURE 11. YOLOv5 + transformer prediction head model.

TABLE 8. YOLOv5 with Transformer Prediction Head performance on pavement damage dataset.

Class	Img	Labels	P	R	mAP@.5	mAP@.95
all	584	1462	0.71	0.672	0.657	0.35
D00	584	349	0.694	0.576	0.551	0.244
D10	584	247	0.734	0.737	0.735	0.34
D20	584	57	0.53	0.439	0.375	0.177
D40	584	627	0.775	0.793	0.791	0.396
Repair	584	182	0.818	0.816	0.832	0.594

ties. Figure 11 shows the YOLOv5 + Transformer Prediction Head model.

The transformer head consists of multiple layers similar to those used in NLP tasks. Each transformer layer contains a self-attention mechanism, which allows the model to attend to different parts of the input features to make predictions. The transformer head also includes fully connected layers and sigmoid activation functions to produce the final output.

The Experiments have shown that the YOLOv5 + Transformer model can achieve higher accuracy than the standard YOLOv5 model while reducing the computational cost of object detection. In the training process, the model takes 212 epochs completed in 6.121 hours, 10 hours less when compared with YOLOv7x-W6. Table 8 shows the performance metrics for object detection using a YOLOv5 model with a transformer head prediction.

The use of transformer architecture allows the model to capture more complex spatial relationships between objects, resulting in better localization and classification performance. Additionally, the transformer-based prediction head can be trained more efficiently, requiring fewer iterations to converge to a good solution. Figures 12 and 13 shows the perfomance of the model.

The YOLOv5 + Transformer model achieved a mAP@.5 of 0.657 on all objects, with a precision of 0.71 and a recall of 0.672. The model also achieved high mAP values for individual classes. Overall, this experiment demonstrates the high performance of the YOLOv5 + transformer model with the dataset.

E. VISUAL ANALYSIS

As shown in Figure 14, YOLOv5 could accurately recognize and identify road damage structures from UAV images. The predicted model outcome closely reflects the initial test photographs, indicating high accuracy and performance. Similarly, YOLOv7's ability to identify road damage structures can be observed in Figures 15, 16 and 17. The model correctly detects the damages, even in cases where the images contain minor flaws or unclear classes.

Visual analysis shows that both YOLOv5 and YOLOv7 can accurately identify and locate road damage structures in UAV images. Specifically, when comparing Figures 14 and 15, it can be seen that YOLOv5 has a slightly more accurate and precise bounding box placement around the damage, whereas YOLOv7 tends to have slightly larger bounding boxes that encompass more of the surrounding area. It can also be observed that YOLOv7 tends to split the damage class into smaller sub-regions, which can be seen in the increased number of bounding boxes around the same area of damage compared to YOLOv5. This is particularly noticeable for the D40 category, which is overrepresented in the data.

In Figure 16, the bounding box around the recheck at the center of the image may be inadequate, possibly due to occlusion or a high box loss value. This highlights the need

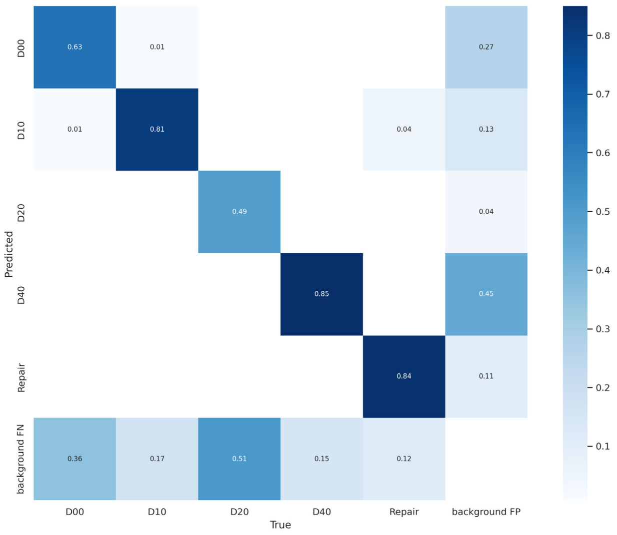


FIGURE 12. Confusion matrix for the YOLOv5 + transformer prediction head model.

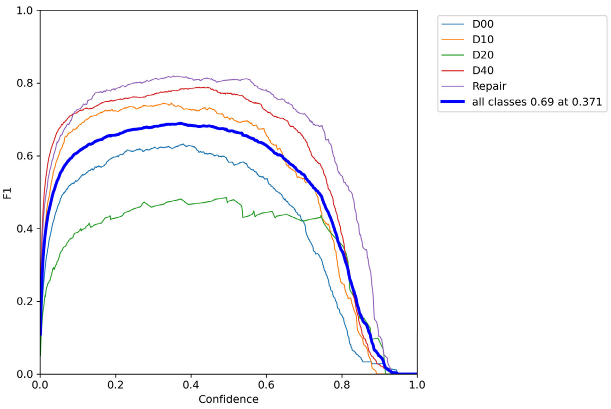


FIGURE 13. F1-Confidence curve.

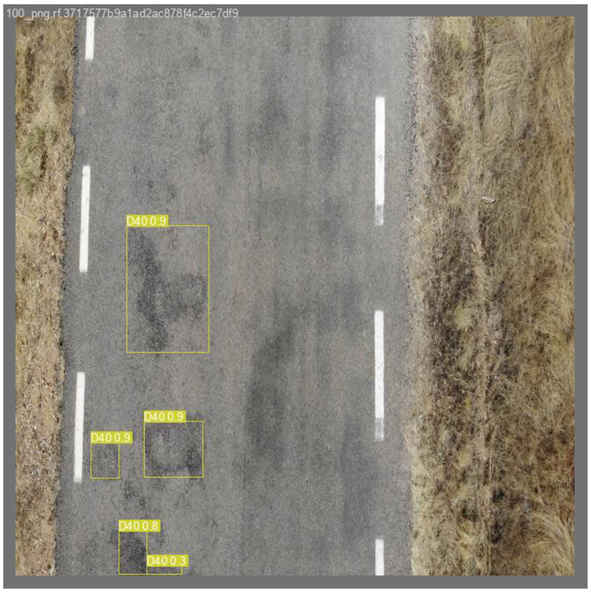


FIGURE 14. Validation YOLOv5.

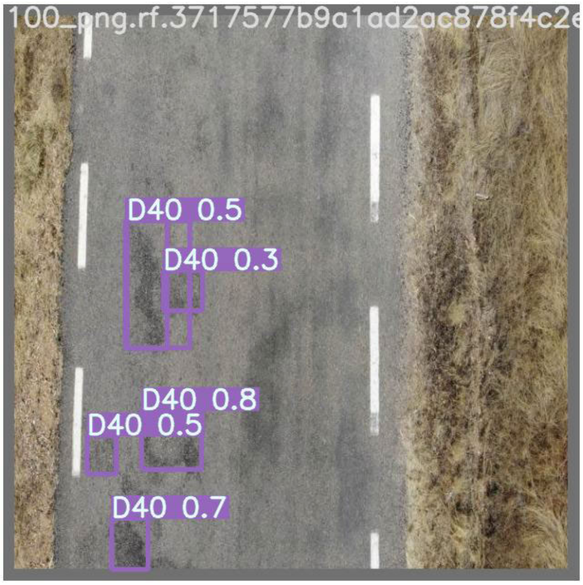


FIGURE 15. Validation YOLOv7.

TABLE 9. Performance comparison for YOLOv4, YOLOv5 and YOLOv7.

	Imgs	P	R	@.5	.5:.95	Time
YOLOv4	584	0.50	0.32	0.26	-	3s
YOLOv5x	584	0.78	0.56	0.59	0.895	17.2ms
YOLOv5 + TPH	584	0.71	0.67	0.65	0.35	6.1ms
YOLOv7	584	0.65	0.78	0.73	0.289	11.4ms

for further improvement in the model's ability to correctly identify and draw bounding boxes around such instances of damage.

F. COMPARISON

Table 9 compares the performance of the YOLOv4, YOLOv5, and YOLOv7 architectures for road damage identification.

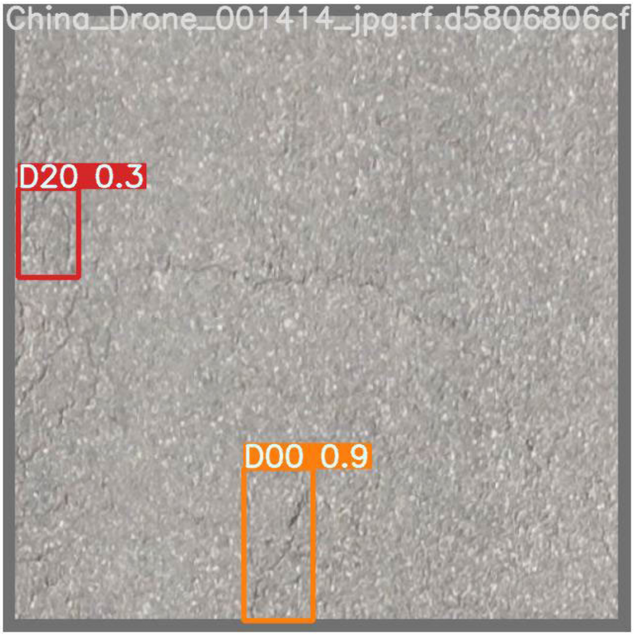


FIGURE 16. Validation YOLOv7.

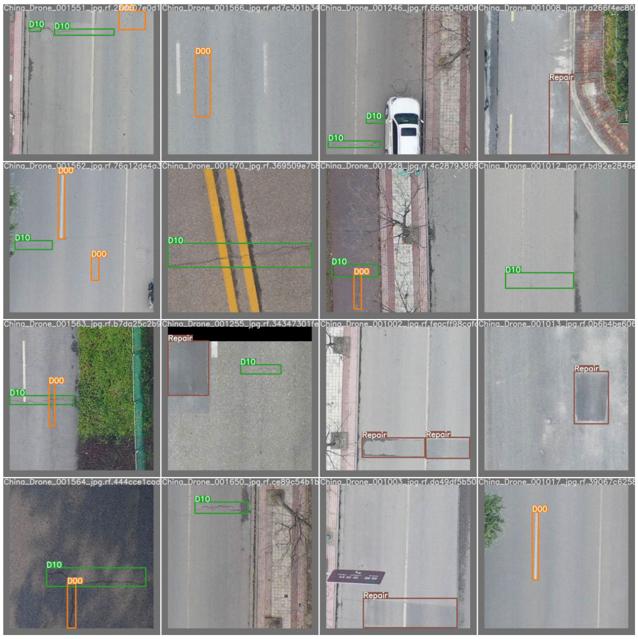


FIGURE 17. Validation YOLOv7X-W6.

The results show that YOLOv4 performed poorly in precision and detection speed, while YOLOv5 and YOLOv7 performed significantly better. Specifically, YOLOv7 stood out for its high accuracy and fast detection time. The table also includes information on the number of images used for testing, precision, recall, mean average precision (mAP), and inference time for each model.

V. CONCLUSION AND FUTURE WORKS

In conclusion, this study compares the YOLOv4 from past work, the YOLOv5 and YOLOv7 architectures, and includes an implementation of the YOLOv5 with Transformer for road damage identification using UAV images. The research successfully achieved its goal of creating an architecture capable of detecting road damage and demonstrated that new architecture versions, such as YOLOv5 and YOLOv7, can improve upon previous work.

A significant contribution of this study was the development of a UAV image database tailored explicitly for training the YOLO versions, which was further enhanced by merging with the RDD2022 dataset. This improved detection of road damage samples, particularly for Spanish and Chinese roads, and helped reduce class imbalance for specific forms of road damage, such as potholes and alligator cracks. The findings of this study provide a valuable contribution to the field and pave the way for future research in this area. As presented in the results section, our implementation achieved a mAP.5 of 26.8% with YOLOv4, 59.9% with YOLOv5, and 73.20% with YOLOv7, finally the implemented Transformer achieved 65.7%. There is still scope for improvement in our work.

Future research can explore the different types of images, such as multispectral images and LIDAR sensors, to further enhance the performance. The fusion of such information is potentially possible to yield better results using embedded computer. Moreover, another approach to this work is the use of fixed-wing UAV.

REFERENCES

[1] H. S. S. Blas, A. C. Balea, A. S. Mendes, L. A. Silva, and G. V. González, "A platform for swimming pool detection and legal verification using a multi-agent system and remote image sensing," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 2023, pp. 1–13, Jan. 2023.

[2] V. J. Hodge, R. Hawkins, and R. Alexander, "Deep reinforcement learning for drone navigation using sensor data," *Neural Comput. Appl.*, vol. 33, no. 6, pp. 2015–2033, Jun. 2020, doi: 10.1007/s00521-020-05097-x.

[3] A. Safonova, Y. Hamad, A. Alekhina, and D. Kaplun, "Detection of Norway spruce trees (*Picea abies*) infested by bark beetle in UAV images using YOLOs architectures," *IEEE Access*, vol. 10, pp. 10384–10392, 2022.

[4] D. Gallacher, "Drones to manage the urban environment: Risks, rewards, alternatives," *J. Unmanned Vehicle Syst.*, vol. 4, no. 2, pp. 115–124, Jun. 2016.

[5] L. A. Silva, A. S. Mendes, H. S. S. Blas, L. C. Bastos, A. L. Gonçalves, and A. F. de Moraes, "Active actions in the extraction of urban objects for information quality and knowledge recommendation with machine learning," *Sensors*, vol. 23, no. 1, p. 138, Dec. 2022, doi: 10.3390/s23010138.

[6] L. Melendy, S. C. Hagen, F. B. Sullivan, T. R. H. Pearson, S. M. Walker, P. Ellis, A. K. Sambodo, O. Roswintiarti, M. A. Hanson, A. W. Klassen, M. W. Palace, B. H. Braswell, and G. M. Delgado, "Automated method for measuring the extent of selective logging damage with airborne LiDAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 139, pp. 228–240, May 2018, doi: 10.1016/j.isprsjprs.2018.02.022.

[7] L. A. Silva, H. S. S. Blas, D. P. García, A. S. Mendes, and G. V. González, "An architectural multi-agent system for a pavement monitoring system with pothole recognition in UAV images," *Sensors*, vol. 20, no. 21, p. 6205, Oct. 2020, doi: 10.3390/s20216205.

[8] M. Guerrieri and G. Parla, "Flexible and stone pavements distress detection and measurement by deep learning and low-cost detection devices," *Eng. Failure Anal.*, vol. 141, Nov. 2022, Art. no. 106714, doi: 10.1016/j.engfailanal.2022.106714.

[9] D. Jeong, "Road damage detection using YOLO with smartphone images," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 5559–5562, doi: 10.1109/BIGDATA50022.2020.9377847.

[10] M. Izadi, A. Mohammadzadeh, and A. Haghighattalab, "A new neuro-fuzzy approach for post-earthquake road damage assessment using GA and SVM classification from QuickBird satellite images," *J. Indian Soc. Remote Sens.*, vol. 45, no. 6, pp. 965–977, Mar. 2017.

[11] Y. Bhatia, R. Rai, V. Gupta, N. Aggarwal, and A. Akula, "Convolutional neural networks based potholes detection using thermal imaging," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 3, pp. 578–588, Mar. 2022, doi: 10.1016/j.jksuci.2019.02.004.

[12] J. Guan, X. Yang, L. Ding, X. Cheng, V. C. Lee, and C. Jin, "Automated pixel-level pavement distress detection based on stereo vision and deep learning," *Automat. Constr.*, vol. 129, p. 103788, Sep. 2021, doi: 10.1016/j.autcon.2021.103788.

[13] D. Arya, H. Maeda, S. K. Ghosh, D. Toshniwal, and Y. Sekimoto, "RDD2022: A multi-national image dataset for automatic road damage detection," 2022, *arXiv:2209.08538*.

[14] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 6517–6525, doi: 10.1109/CVPR.2017.690.

[15] J. Redmon and A. Farhadi. *YOLOv3: An Incremental Improvement*. [Online]. Available: https://pjreddie.com/yolo/

[16] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[17] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, Y. Kwon, K. Michael, J. Fang, C. Wong, D. Montes, Z. Wang, C. Fati, J. Nadar, V. Sonck, P. Skalski, A. Hogan, D. Nair, M. Strobel, and M. Jain, "Ultralytics/YOLOv5: V7.0—YOLOv5 SOTA realtime instance segmentation," Zenodo, Tech. Rep., Nov. 2022. [Online]. Available: https://zenodo.org/record/7347926

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

[19] R. Ali, D. Kang, G. Suh, and Y.-J. Cha, "Real-time multiple damage mapping using autonomous UAV and deep faster region-based neural networks for GPS-denied structures," *Autom. Construct.*, vol. 130, Oct. 2021, Art. no. 103831. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092658052100282X

[20] D. Kang and Y.-J. Cha, "Autonomous UAVs for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging: Autonomous UAVs for SHM," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 33, no. 10, pp. 885–902, Oct. 2018.

[21] Z. Xu, H. Shi, N. Li, C. Xiang, and H. Zhou, "Vehicle detection under UAV based on optimal dense YOLO method," in *Proc. 5th Int. Conf. Syst. Informat. (ICSAI)*, Nov. 2018, pp. 407–411, doi: 10.1109/ICSAI.2018.8599403.

[22] P. Kannadaguli, "YOLO v4 based human detection system using aerial thermal imaging for UAV based surveillance applications," in *Proc. Int. Conf. Decis. Aid Sci. Appl. (DASA)*, Nov. 2020, pp. 1213–1219, doi: 10.1109/DASA51403.2020.9317198.

[23] T. Petso, R. S. Jamisola, D. Mpoeleng, and W. Mmereki, "Individual animal and herd identification using custom YOLO v3 and v4 with images taken from a UAV camera at different altitudes," in *Proc. IEEE 6th Int. Conf. Signal Image Process. (ICSIP)*, Oct. 2021, pp. 33–39, doi: 10.1109/ICSIP52628.2021.9688827.

[24] L. Wang and Z. Zhang, "Automatic detection of wind turbine blade surface cracks based on UAV-taken images," *IEEE Trans. Ind. Electron.*, vol. 64, no. 9, pp. 7293–7303, Sep. 2017, doi: 10.1109/TIE.2017.2682037.

[25] D. Sadykova, D. Pernebayeva, M. Bagheri, and A. James, "IN-YOLO: Real-time detection of outdoor high voltage insulators using UAV imaging," *IEEE Trans. Power Del.*, vol. 35, no. 3, pp. 1599–1601, Jun. 2020, doi: 10.1109/TPWRD.2019.2944741.

[26] M. A. A. Khan, M. Alsawwaf, B. Arab, M. AlHashim, F. Almashharawi, O. Hakami, S. O. Olatunji, and M. Farooqui, "Road damages detection and classification using deep learning and UAVs," in *Proc. 2nd Asian Conf. Innov. Technol. (ASIANCON)*, Aug. 2022, pp. 1–6, doi: 10.1109/ASIANCON55314.2022.9909043.

[27] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 5, pp. 361–378, May 2017.

[28] M. Böyük, R. Duvar, and O. Urhan, "Deep learning based vehicle detection with images taken from unmanned air vehicle," in *Proc. Innov. Intell. Syst. Appl. Conf. (ASYU)*, Oct. 2020, pp. 1–4, doi: 10.1109/ASYU50717.2020.9259868.

[29] R. Li, J. Yu, F. Li, R. Yang, Y. Wang, and Z. Peng, "Automatic bridge crack detection using unmanned aerial vehicle and faster R-CNN," *Construct. Building Mater.*, vol. 362, Jan. 2023, Art. no. 129659, doi: 10.1016/j.conbuildmat.2022.129659.

[30] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 33, no. 9, pp. 731–747, Sep. 2018.

[31] D. Kang, S. S. Benipal, D. L. Gopal, and Y.-J. Cha, "Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning," *Autom. Construct.*, vol. 118, Oct. 2020, Art. no. 103291.

[32] S. Shim, J. Kim, S.-W. Lee, and G.-C. Cho, "Road damage detection using super-resolution and semi-supervised learning with generative adversarial network," *Autom. Construct.*, vol. 135, Mar. 2022, Art. no. 104139, doi: 10.1016/j.autcon.2022.104139.

[33] D. H. Kang and Y.-J. Cha, "Efficient attention-based deep encoder and decoder for automatic crack segmentation," *Struct. Health Monitor.*, vol. 21, no. 5, pp. 2190–2205, Sep. 2022.

[34] R. Ali and Y.-J. Cha, "Attention-based generative adversarial network with internal damage segmentation using thermography," *Autom. Construct.*, vol. 141, Sep. 2022, Art. no. 104412.

[35] W. Choi and Y. Cha, "SDDNet: Real-time crack segmentation," *IEEE Trans. Ind. Electron.*, vol. 67, no. 9, pp. 8016–8025, Sep. 2020.

[36] D. Arya, H. Maeda, S. Kumar Ghosh, D. Toshniwal, H. Omata, T. Kashiyama, and Y. Sekimoto, "Global road damage detection: State-of-the-art solutions," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 5533–5539, doi: 10.1109/BIGDATA50022.2020.9377790.

[37] V. Pham, C. Pham, and T. Dang, "Road damage detection and classification with Detectron2 and faster R-CNN," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 5592–5601, doi: 10.1109/BIGDATA50022.2020.9378027.

[38] L. Parameswaran, "Deep learning based detection of potholes in Indian roads using YOLO," in *Proc. Int. Conf. Inventive Comput. Technol. (ICICT)*, Feb. 2020, pp. 381–385, doi: 10.1109/ICICT48043.2020.9112424.

[39] Y. Liu, G. Shi, Y. Li, and Z. Zhao, "M-YOLO based detection and recognition of highway surface oil filling with unmanned aerial vehicle," in *Proc. 7th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Apr. 2022, pp. 1884–1887, doi: 10.1109/ICSP54964.2022.9778782.

[40] Y. O. Ouma and M. Hahn, "Pothole detection on asphalt pavements from 2D-colour pothole images using fuzzy *c*-means clustering and morphological reconstruction," *Autom. Construct.*, vol. 83, pp. 196–211, Nov. 2017, doi: 10.1016/j.autcon.2017.08.017.

[41] M. Abdellatif, H. Peel, A. G. Cohn, and R. Fuentes, "Pavement crack detection from hyperspectral images using a novel asphalt crack index," *Remote Sens.*, vol. 12, no. 18, pp. 1–10, 2020. [Online]. Available: https://www.mdpi.com/2072-4292/12/18/3084

[42] F. Viel, R. C. Maciel, L. O. Seman, C. A. Zeferino, E. A. Bezerra, and V. R. Q. Leithardt, "Hyperspectral image classification: An analysis employing CNN, LSTM, transformer, and attention mechanism," *IEEE Access*, vol. 11, pp. 24835–24850, 2023.

[43] D. Arya, H. Maeda, S. K. Ghosh, D. Toshniwal, H. Omata, T. Kashiyama, and Y. Sekimoto, "Crowdsensing-based road damage detection challenge (CRDDC-2022)," 2022, *arXiv:2211.11362*.

LUÍS AUGUSTO SILVA (Member, IEEE) is currently pursuing the Ph.D. degree in computer engineering with the University of Salamanca, Spain. He is a Researcher with the Expert Systems and Applications Laboratory (ESALab). His research interests include object detection, image segmentation, deep learning applied to aerial/satellite images, the Internet of Things, and embedded UAV systems.

VALDERI REIS QUIETINHO LEITHARDT (Senior Member, IEEE) received the Ph.D. degree in computer science from INF-UFRGS, Brazil, in 2015. He is currently a Professor with the Polytechnic Institute of Portalegre and a Researcher integrated with the VALORIZA Research Center for Endogenous Resource Valorization. He is also a Collaborating Researcher with the Expert Systems and Applications Laboratory (ESALab), University of Salamanca, Spain. His main research interests include distributed systems, focusing on data privacy, communication, and programming protocols, involving scenarios and applications for the Internet of Things, smart cities, big data, cloud computing, and blockchain.

VIVIAN FÉLIX LÓPEZ BATISTA received the Ph.D. degree in computer science from the University of Valladolid, in 1996. Since 1998, she has been a Full Professor of computer science with the University of Salamanca, Spain. She performed a research stay with the Big Data Analytics and Data Mining Program, Center for Computational Science, University of Miami, in 2012. She was the Director of the Master's in Intelligent System and Program of Ph.D. in computer science with the University of Salamanca, from 2009 to 2012. She is currently a member of the Data Mining Group (MIDA). She has done research on natural language processing, machine learning, and neural networks. Furthermore, she has also papers published in recognized journals, workshops and symposiums, books, and book chapters on these topics. She has two six-year of research granted the first one, in June 2013, and the second one covering the evaluation period, from 2012 to 2018, was granted in 2019. She has a total of 25 JCR. Her average number of citations/year during the last five years is 18.4. According to Mendeley, her H-index: 17. She has supervised Ph.D. theses within the Doctoral Program, University of Salamanca. She has been a member of the organizing and scientific committee of several international symposiums.

GABRIEL VILLARRUBIA GONZÁLEZ received the master's degree in intelligent systems from the University of Salamanca, in 2012, the master's degree in internet security, in 2014, the master's degree in information systems management, in 2015, and the Ph.D. degree from the Department of Computer Science and Automation, University of Salamanca. He was a Computer Engineer with the Pontifical University of Salamanca, in 2011. He is currently an Associate Professor with the University of Salamanca and a Researcher with the Expert Systems and Applications Laboratory (ESALab). Throughout his training, he has followed a well-defined line of research focused on applying multi-agent systems to ambient intelligence environments, particularly concerning the definition of intelligent architectures and information fusion.

JUAN FRANCISCO DE PAZ SANTANA received the degree in technical engineering in systems computer sciences, the degree in engineering in computer sciences, the degree in statistics, and the Ph.D. degree in computer science from the University of Salamanca, Spain, in 2003, 2005, 2007, and 2010, respectively. He is currently a Full Professor with the University of Salamanca and a Researcher with the Expert Systems and Applications Laboratory (ESALab). He is the coauthor of published papers in several journals, workshops, and symposiums.