A Foveated Approach to Automated Billboard Detection

Sayali Avinash Chavan Dermot Kerr Sonya Coleman Hussein Khader*
Intelligent Systems Research Centre
University of Ulster
Londonderry, United Kingdom
**The Neuron*, Amman, Jordan
email:{chavan-s, d.kerr, sa.coleman}@ulster.ac.uk, hussein.khader@theneuron.com

***Abstract*—Understanding billboard visibility is vital when considering the value of each billboard to advertisers, hence the growing demand for artificial intelligence based approaches to visibility measurement. Addressing this need, this research paper presents a comprehensive approach to billboard detection using street-view images. We have developed a robust billboard detection system by leveraging state-of-the-art object detection models, such as You Only Look Once (YOLOv8), YOLOv5, Faster-Region-based Convolutional Neural Network (RCNN) and CenterNet resulting in high model accuracy. We have introduced an innovative foveated approach, based on the human visual systems, that applies a Gaussian function to assign weights to billboards to determine which is the most significant billboard based on a combination of confidence and location with respect to the image centre. The approach demonstrates an improvement in overall accuracy of the detection process. In particular YOLOv8 experienced a high accuracy increase from 63.40 to 82.71 percent. This research provides valuable insights and practical solutions for billboard detection in real-time.**

Index Terms*—Object Detection; Deep Learning; YOLO; Convolutional Neural Network.

I. Introduction

Billboards play a significant role in outdoor advertising, aiming to capture the attention of individuals and deliver brand messages effectively. Understanding the visibility of a billboard can help advertisers place value on those which are more desirable considering the likelihood that the advertisement can be clearly observed. Detecting billboards automatically with computer vision can provide valuable insights for this purpose. It allows advertisers and marketers to assess the impact and reach of their advertisement campaigns. It also helps them to evaluate the effectiveness of their strategies and facilitates data driven decision-making tools. Such insights can guide businesses in making informed decisions and optimise their advertising strategies for maximum impact and return on investment. With the ever-increasing presence of billboards in urban areas, the need to detect and analyse billboard visibility has become essential [1].

However, accurately identifying billboards remains a challenging task due to their diverse shapes, sizes, types and the environment within which they reside. Additionally, occlusions, obstructions, and lighting variations further complicate the detection process for the network. Developing a universal billboard detection algorithm that works accurately across different types of billboards requires training on a diverse dataset for robust detection [2]. Hence, in this paper, we present a novel approach to billboard detection using state-of-the-art object detection models, combined with a function that prioritises billboards in desired locations.

This paper makes significant contributions in the field of billboard detection using street-view images with a comprehensive dataset, development of a robust detection algorithm and an approach to prioritising billboards dependent on their location within a scene. Through extensive experimentation and evaluation, we demonstrate the effectiveness of this approach in achieving higher accuracy in comparison to popular models, such as CenterNet [3], Faster-RCNN [4], YOLOv5 [5] and latest version of YOLO [6]- YOLOv8 [7]. The results indicate that our proposed method achieves significant performance using data in the wild, reaching 82.71% correctly detected billboards on unseen data. Our work offers valuable insights and practical solutions that can be utilised in various applications, including urban planning, advertising analysis, and outdoor media management.

This paper is structured as follows: In Section II, we examine the state-of-the-art in billboard detection, identifying key gaps that our study aims to fill with new insights. We outline our chosen model, dataset, and training process in Section III, followed by the implementation details in Section IV which highlights our efforts to enhance each model through hyperparameter tuning, showcasing the refinement of each network architecture. The introduction of the Gaussian weighting algorithm further improves model accuracy. Finally, in Section V, we present the outcomes of our study through an extensive analysis of the results, providing valuable insights for future research in this field, as summarised in Section VI.

II. State of Art

For object detection, feature-based detection methods widely utilise approaches such as Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG), Speeded-Up Robust Features (SURF), and Oriented FAST and Rotated BRIEF (ORB) [8]. However, these methods possess inherent limitations. They have low detection rates,

 ISBN: 978-1-68558-115-2

are sensitive to changes in illumination and struggle with complex backgrounds, occlusions, and variations in rotation. Moreover, these approaches are slow and encounter difficulties when dealing with multiple objects in low-quality images. As a result, Convolutional Neural Network (CNN) have emerged as a popular solution, offering high-speed object detection and recognition [9]. CNNs have revolutionised the field by enabling automatic feature learning directly from raw image data. Specifically designed CNN models for object detection can effectively detect patterns in image data and leverage transfer learning for improved performance [10].

Over the past two decades, numerous ground breaking object detection models have been published and extensively studied [11]. These models include SSD (Single Shot Detector) [12], CenterNet [3], RCNN [13], Fast RCNN [14], Faster RCNN [4], YOLO [6], Feature Pyramid Network (FPN) [15], Retina-Net [16], RefineDet [17], Spatial Pyramid Pooling Network (SPPNet) [18], Deformable Part-based Model (DPM) [19], TridentNet [20], Fully Convolutional One-Stage Detector (FCOS) [21], Hybrid Task Cascade (HTC) [22], Deformable DETR [23], and many more. Recent cutting-edge advancement in the field of image segmentation is the Segment Anything Model (SAM) developed by Meta AI Research [24]. SAM is an instance segmentation model trained on an extensive dataset comprising 11 million images and 1.1 billion segmentation masks. Its state-of-the-art performance makes it highly accurate for real-time applications. However, despite these recent developments, the YOLO model remains widely popular in object detection applications due to its real-time high accuracy [6] [11]. YOLO has evolved from its initial version, YOLOv1, to the most recent version, YOLOv8 [5] [7]. YOLOv8 is capable of performing segmentation, classification, detection, tracking, and pose detection, making it a versatile and powerful model in the field.

For the specific application of billboard detection, previous studies have primarily concentrated on detecting advertising billboards using techniques, such as edge detection and planar object detection [25] [26]. However, these researches encounter localisation difficulties when confronted with multiple objects present in the scene. Additionally, research has targeted billboards specifically in soccer fields and sport TV broadcasts, employing methods like the Fast Fourier Transform and Hough transforms [27] [28]. Yet, it is crucial to acknowledge that their ability to accurately detect billboards was reliant on the use of a high-accuracy camera capable of capturing high-resolution pictures.

Further research has been conducted in the field of billboard detection, with a particular emphasis on the text and content displayed within advertisements. One machine learning-based approach aimed to identify illegal advertisements by analysing the extracted content from the advertisements themselves [29]. In another study, the detection of photo manipulation on billboards was accomplished using a novel forensic technique that assessed adherence to the rules of perspective projection [30]. Additionally, a study employed Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) to recognise text present on billboards [31]. Compliance-related research explored the use of computer vision techniques to identify advertisements on buildings, as demonstrated by [32]. The billboard detection examples in [29]–[32] primarily focuses on text or content recognition, lacking a comprehensive approach to detect billboards regardless of their content. Therefore, in our research, we aimed to delve into the detection process irrespective of the billboard's content.

Shifting the focus to video frames, the development of the ADNet architecture allowed for the detection of advertising instances, with the utilisation of the Microsoft COCO (Common Objects in Context) dataset for training purposes [33]. A significant drawback of ADNet is its lack of scalability, which raises concerns about its ability to perform effectively in diverse real-world scenarios.

Research by Liu et al. [34] introduced combining attention-based multi-scale features with Faster RCNN for the purpose of billboard detection. They acknowledged billboard detection challenges such as small object sizes, cluttered backgrounds, and low resolutions impose limitations on accuracy improvement. A comparative study regarding urban billboard detection comparing SSD and YOLO models exhibited promising results [35]. However, both research studies in [34] and [35] demonstrated satisfactory performance within the constraints of their respective high-resolution limited datasets. Furthermore, the research conducted by Chavan et al. [2] demonstrated successful outcomes in billboard classification and detection. Overall, the implementation of state of art object detection models shows promise in overcoming real-world application challenges.

III. METHODOLOGY

The methodology section provides a comprehensive outline of the research approach employed in this study, offering a clear roadmap for model selection, data collection, data cleaning and training a custom model using a dataset containing ground truth bounding boxes for billboards. We fine-tune the models using a large number of images, annotations and hyperparameters. To further enhance the accuracy, we apply a Gaussian distribution-based weighting [36] to the centre of the detected objects during testing to deal with real world challenges such as multiple billboards within a scene.

A. Model Selection

After conducting extensive research on the current state-of-the-art models [11], we carefully selected four networks: Faster RCNN, CenterNet, YOLOv5, and YOLOv8.

- Faster RCNN stands out for its exceptional accuracy in object detection. By utilising a region proposal network, it generates potential object locations and then fine tunes the model for improved localisation and recognition, which is essential for applications requiring reliable results [4].
- CenterNet focuses on estimating the centre points of objects. It can accurately determine the position of the object, enabling precise centre detection and localisation of billboards. Its centre point estimation approach allows

 ISBN: 978-1-68558-115-2

it to handle and distinguish multiple objects efficiently, making it suitable for scenarios where multiple billboards may be present in the scene [3].

- YOLO is renowned for its impressive speed in object detection. It can process images in real-time. Moreover, YOLOv8, the latest version released in 2023, is designed to be computationally efficient enabling faster processing, which can be advantageous in scenarios where quick detection is crucial [6].

B. Image Dataset

We obtained longitudinal and latitudinal coordinates from an Out of Home (OOH) industry partner [37] for billboard locations in the United Kingdom, which were used as reference points for retrieving corresponding street-view images programmatically. A meticulous cleaning process was completed to ensure the dataset's quality and relevance. This involved removing duplicate images and filtering any irrelevant or low-quality images (as depicted in Figure 1) that did not accurately represent the billboard locations. After cleaning the dataset, we have 3,437 images (examples of which are shown in Figure 2), which were subsequently divided into three subsets: a training set comprising 2,500 images, a validation set consisting of 700 images, and a test set containing 238 images. The split of approximately 73% training, 20% validation, and 7% test optimises a substantial training dataset for model training, a validation set for precise hyperparameter tuning, and a carefully selected test set for an unbiased final evaluation [38].

The dataset consists of images obtained from various locations across the UK, with different types of billboards, including digital and static billboards, street furniture, spectacular billboards, and illuminated billboards. The billboards exhibit a wide range of designs, layouts, and content, showcasing promotional messages for products, services and events. Furthermore, the images also capture contextual elements such as trees, buildings, roads, pedestrians, and vehicles illustrates the difficulty in detection. By following this comprehensive data extraction process - created the custom billboard dataset, which involved obtaining billboard coordinates, extracting street view images surrounding each billboard to obtain an image dataset, cleaning and dividing the data and performing manual annotation. The manual annotation step entailed reviewing each image and accurately labeling the billboards by placing bounding boxes around them.

C. Training Process

The training process involves optimising the models parameters and weights through an iterative process to ensure accurate detection and localisation of billboards. The models (Faster RCNN, CenterNet, YOLOv5, and YOLOv8) were fine-tuned using the annotated dataset, allowing them to learn and adapt to the specific characteristics of billboard images in the UK regions. This is the general flow of the system:

- Input images - fed into the object detection model.
- Loss function - calculated based on the model predictions and ground truth.

Fig. 1. Example of a low-quality or irrelevant image (blurry image) filtered during dataset cleaning process.

Fig. 2. Training sample from image dataset for UK region showcasing billboards in real-life environments with surrounding vehicles and background scenery.

- Model parameters - updated using gradient descent optimisation, minimising the loss.
- The training process iterates over the dataset multiple times depending on the number of steps/epochs to improve the model's performance.

This comprehensive training process enabled the models to achieve superior performance and effectiveness in detecting and recognising billboards in various real-life scenarios.

We trained Faster RCNN, CenterNet, and two versions of YOLO; these were compared based on their accuracy. We analysed their training and testing accuracy, and robustness to varying billboard images with single and multiple billboards in one image. In Section V, we conducted an evaluation of the models, focusing on their ability to generalise to unseen urban scenes (test set of 238 images).

IV. Model Implementation

The following section elaborates on the practical implementation of each network, elucidating the architectural details employed in this research.

A. Faster RCNN

Faster RCNN is a popular framework for object detection in computer vision. It consists of a convolutional neural network (CNN) backbone, such as ResNet or VGGNet, for feature extraction. As shown in Figure 3, Faster RCNN combines a Region Proposal Network (RPN) with a CNN-based object detection network to efficiently detect objects in an image.

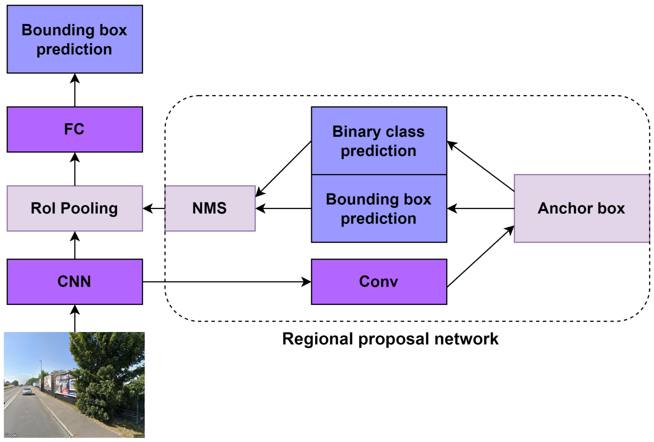

Fig. 3. Faster RCNN Architecture.

The RPN generates a set of region proposals. It generates candidate object proposals, which are then refined using a Region of Interest (RoI) pooling layer and passed through fully connected layers for classification and bounding box regression. RPN then shares full-image convolutional features with a detection network, thus enabling nearly cost-free region proposals. Faster RCNN merges the RPN and Fast RCNN into a single network by sharing their convolutional features 'attention' mechanism. The RPN component tells the unified network where to look. The key steps involved in an RPN for object detection framework cover the following aspects in sequence:

- Input: Image of size $W \times H$
- Convolutional feature map: $F = CNN(Image)$
- Anchor generation: Generate a set of fixed-size anchor boxes at different scales and aspect ratios.
- Anchor classification: For each anchor, predict the probability of it containing an object (foreground) or not (background).
- Anchor regression: For each foreground anchor, refine the coordinates of the bounding box to better fit the object.

The Object Detection Network processes RoIs and performs object classification and bounding box regression. Convolutional feature maps can be used for generating region proposals. This is constructed by adding a few more convolutional layers so that the model performs both localisation and regression tasks at the same time, thus a FCN (Fully Convolutional Neural Network) can be trained end to end specifically for the task of generating detection proposals. The main components of an object detection framework encompass the following aspects in sequence:

- Input: Convolutional feature map F from the RPN and region proposals.
- RoI pooling: Crop and resize the features within each region proposal to a fixed size.
- Fully connected layers: Pass the RoI-pooled features through a series of fully connected layers.
- Classification: Predict the class probabilities for each region proposal.
- Bounding box regression: Refine the coordinates of the bounding boxes.

In the framework, the loss function is used to train the model and optimise the network parameters. The loss function consists of two components: the classification loss (L_{cls}) and the regression loss (L_{reg}). The overall loss for the RPN is denoted as L_{rpn} and defined as the sum of the classification loss and the regression loss, with the regression loss multiplied by a balancing parameter (keeping λ = 10) [4]:

$$L_{\text{rpn}} = L_{\text{cls}}(p, t) + \lambda \cdot L_{\text{reg}}(t^b, t_i^b) \quad (1)$$

The classification loss (L_{cls}) is computed for each anchor and averaged over the number of classes (N_{cls}):

$$L_{\text{cls}} = \frac{1}{N_{\text{cls}}} \sum_c L_{\text{cls}}(p_c, t_c) + \lambda \cdot L_{\text{reg}}(t_b, t_i^b) \quad (2)$$

Here, t_i is the ground truth label for the anchor, t_i^b is the ground truth bounding box regression targets, $L_{\text{cls}}(p_c, t_c)$ is the binary logistic loss between the predicted probability p_c and the ground truth label t_c, and $L_{\text{reg}}(t_b, t_i^b)$ is the smooth $L1$ loss for bounding box regression. The term λ is a balancing parameter that determines the relative importance of the classification and regression components in the overall loss. By minimising this loss function during training, the Faster RCNN model learns to generate accurate region proposals and accurately classify and localise objects within those regions. Faster RCNN achieves impressive accuracy but is relatively slow compared to some newer models [39].

B. CenterNet

CenterNet is an innovative and efficient single-shot object detection model that deviates from traditional methods by focusing on predicting the center points of objects instead of explicitly predicting bounding boxes. By regressing the center point and object size, CenterNet achieves high accuracy and efficiency. Unlike other object detectors that rely on bounding box estimation, CenterNet adopts a keypoint estimation approach. It detects each object as a triplet of keypoints, specifically the object's center point and the two corners of its bounding box.

The CenterNet architecture comprises a backbone network, intermediate heatmaps for keypoint estimation, and offset regression maps for bounding box prediction, as shown in Figure 4. It incorporates two custom modules: cascade corner pooling and center pooling. Cascade corner pooling enriches information gathered from the top-left and bottom-right corners of objects, while center pooling provides more notable information from the central regions. This combination enhances the model's ability to capture both corner and center characteristics. The output includes corner heatmaps and center heatmaps, indicating the likelihood of object corners and

 ISBN: 978-1-68558-115-2

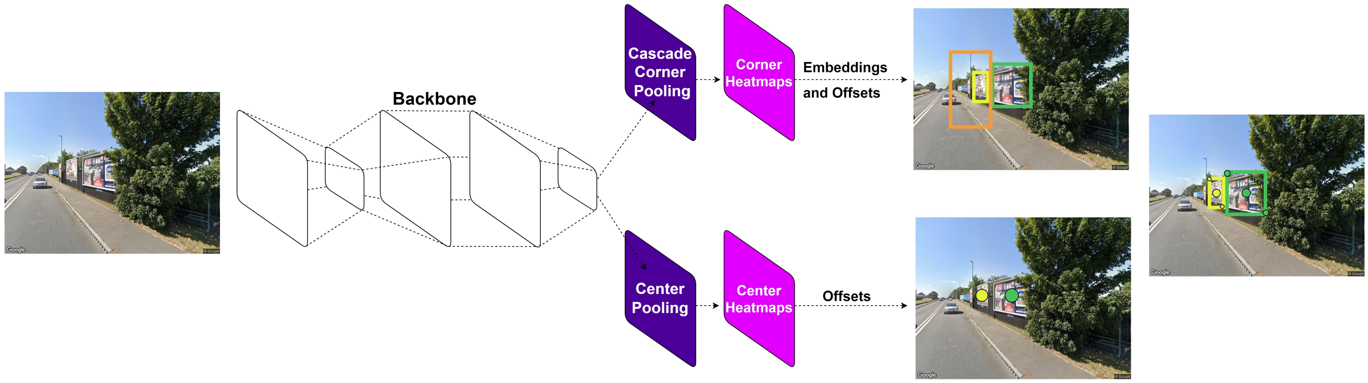

Fig. 4. CenterNet structure.

centers at different spatial locations. Additionally, embedding offsets refine localisation by providing displacement information. The final step involves localising the object center using the corner heatmaps, center heatmaps, and embedding offsets.

During inference, CenterNet utilises the predicted corner keypoints to generate proposals, and then checks if there is a center keypoint of the same class falling within the central region of each proposal. If a centre keypoint of the same class is found, it confirms the proposal as a valid object.

The loss function in CenterNet comprises the heatmap loss, which measures the dissimilarity between predicted and ground truth heatmaps using binary cross-entropy. Regression loss refines bounding box coordinates using the smooth $L1$ loss. The total loss combines the heatmap loss and regression loss, with a balancing parameter to control their relative importance. Mathematically, the loss function in CenterNet can be summarised as [3]:

$$L_{\text{total}} = L_{\text{heatmap}} + \lambda \cdot L_{\text{reg}} \tag{3}$$

where L_{heatmap} represents the binary cross-entropy loss between predicted and ground truth heatmaps, and L_{reg} denotes the smooth L1 loss for refining bounding box coordinates. λ is the balancing parameter [40].

CenterNet achieves accurate object detection while maintaining efficiency in terms of computational resources and processing time [3].

C. YOLO

YOLO is a real-time object detection framework that aims to achieve high detection accuracy with fast inference speed. It divides the input image into a grid and predicts bounding boxes and class probabilities directly. The YOLO architecture [41] employs a CNN backbone (as shown in Figure 5), followed by a series of convolutional layers. These layers simultaneously predict the bounding box coordinates, objectness score, and class probabilities at multiple grid scales, allowing for accurate and efficient object detection. YOLO can be described as a deformable parts model that utilises a sliding window approach. The network performs multiple tasks concurrently, including feature extraction, bounding box prediction, non-maximal suppression, and contextual reasoning. This unified approach contributes to its efficiency and real-time object detection capabilities [6].

In contrast to traditional methods like RCNN, YOLO uses grid cells to propose potential bounding boxes and scores for objects. However, YOLO applies spatial constraints to these grid cell proposals, which helps mitigate multiple detections and leads to far fewer bounding box proposals. One of the notable advantages of YOLO is that it is a unified model for object detection. It can be directly trained on full images, unlike classifier-based approaches. YOLO models, such as YOLOv3 and YOLOv4, have shown good performance on various object detection benchmarks. We have chosen to use both YOLOv5 and YOLOv8 for billboard detection [42].

YOLOv5 has gained widespread recognition and popularity in the field of object detection. It stands out for its exceptional performance, achieving good results in terms of accuracy and speed using various benchmark datasets. This makes it a highly promising choice for object detection tasks. One significant advantage of YOLOv5 is its efficiency and lightweight nature. It utilises a CSPDarknet53 backbone, which enhances feature extraction capabilities and contributes to improved detection accuracy. The YOLOv5 network architecture consists of 20 convolutional layers, followed by an average-pooling layer and a fully connected layer. By incorporating both convolutional and connected layers, the ImageNet pre-trained YOLOv5 model has shown improved performance in object detection tasks [43].

YOLOv8 is the latest version of the YOLO model. Although it shares the same architecture as its predecessors, it introduces several improvements. These improvements include a new neural network architecture that combines the Feature Pyramid Network (FPN) and the Path Aggregation Network (PAN). The FPN in YOLOv8 gradually reduces the spatial resolution of the input image while increasing the number of feature channels. This results in the creation of feature maps that can effectively detect objects at different scales and resolutions. On the other hand, the PAN architecture aggregates features from multiple levels of the network using skip connections. By doing so, the network can capture features at various scales and resolutions

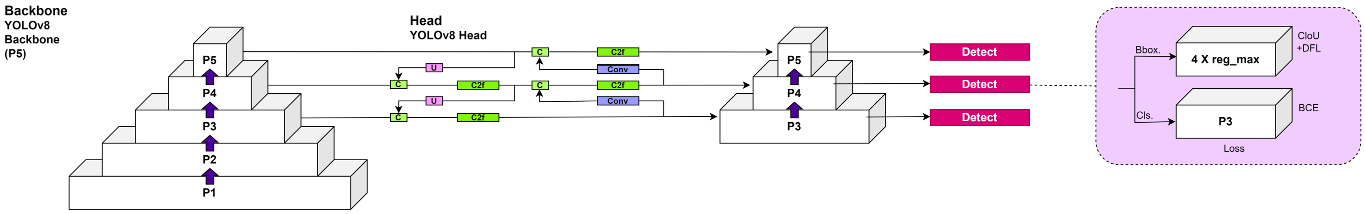

Fig. 5. YOLOv8 Architecture.

more comprehensively, which is vital for accurate detection of objects with different sizes and shapes [41].

The loss function in YOLO guides the optimisation process by aligning predicted bounding box coordinates, improving object localisation accuracy, and encouraging accurate object presence estimation and class prediction. The balancing parameters λ_{cls} , λ_{loc}, λ_{obj} provides the flexibility to fine-tune the significance of different components in the loss function. This allows for optimising the model's performance during training by adjusting the importance given to each specific loss term. The loss function helps the model learn and improve its object detection capabilities, leading to more accurate predictions during inference than predictions made without utilising the loss function [7].

$$L_{\text{yolo}} = \lambda_{\text{loc}} \cdot L_{\text{loc}} + \lambda_{\text{obj}} \cdot L_{\text{obj}} + \lambda_{\text{cls}} \cdot L_{\text{cls}} \tag{4}$$

where L_{loc} is the localisation loss, L_{obj} is the objectness loss, and L_{cls} is the classification loss.

D. Hyper-Parameters

It is important to note that the selection of hyperparameters is a crucial aspect of training deep learning models. The chosen values are influenced by factors such as the dataset characteristics, model architecture, available computational resources, and desired trade-off between speed and accuracy. Faster RCNN, CenterNet, YOLOv5 and YOLOv8 default settings served as a starting point for our experiments, and we further fine-tuned the hyperparameters to optimise the model's performance on our specific dataset using the values shown in Table I. The batch size is the number of samples processed at once during training. Larger batch sizes can lead to faster convergence, but it can also require more memory. The step size decides the total number of training iterations. Whereas the learning rate determines how quickly model learns and updates its parameters during optimisation. It affects both the speed at which the model converges and the accuracy achieved during the training process. An optimizer is a mathematical algorithm used in machine learning to adjust the parameters of a model in order to minimise an error function or maximise a desired output. It plays a vital role in improving the performance and efficiency of models. The following optimizers have been used:

TABLE I
HYPERPARAMETER

Model	Faster RCNN	CenterNet	YoloV5	YoloV8
Image Size	640	512	416	800
Batch Size	1	1	32	16
Optimizer	Momentum	Adam	SGD	SGD
Step Size / Epoch	0-2000	0-2000	0-100	0-50
Warm up Learning Rate	0.001	0.001	0.001	0.001
Step Size / Epoch	2000-25,000	2000-25,000	0-100	0-50
Final Learning Rate	0.004	0.004	0.001	0.001

- Momentum optimizer: The momentum optimizer incorporates a momentum term to accelerate convergence by considering the accumulated velocity from previous updates. The equations for the momentum optimizer are as follows:

$$v_t = \gamma \cdot v_{t-1} + \alpha \cdot \nabla J(W_t) \tag{5}$$

$$W_{t+1} = W_t - v_t \tag{6}$$

Here, v_t represents the velocity at time step t, γ is the momentum coefficient, α is the learning rate, $\nabla J(W_t)$ is the gradient of the loss function with respect to the weights W_t, and W_{t+1} is the updated weights [44].

- Adam optimizer: The Adam optimizer combines concepts from Momentum and root mean square propagation (RMSprop), adapting the learning rate for each parameter based on past gradients. The equations for the Adam optimizer are as follows:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot \nabla J(W_t) \tag{7}$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot (\nabla J(W_t))^2 \tag{8}$$

$$W_{t+1} = W_t - \frac{\alpha}{\sqrt{v_t} + \epsilon} \cdot m_t \tag{9}$$

Here, m_t represents the first moment estimate (mean) at time step t, β_1 is the decay rate for the first moment estimate, m_{t-1} is the first moment estimate at time step $t-1$, v_t represents the second moment estimate (variance) at time step t, β_2 is the decay rate for the second moment estimate, $\nabla J(W_t)$ is the gradient of the loss function with respect to the weights W_t, W_t is the weights at time step t, W_{t+1} is the updated weights at time step $t+1$, α is the learning rate, and ϵ is a small value for numerical stability [44].

- Stochastic Gradient Descent (SGD) optimizer: SGD is a basic optimization algorithm that updates the parameters based on the gradient of the loss function computed using

Copyright (c) IARIA, 2023. ISBN: 978-1-68558-115-2

a small subset (batch) of training data. The equation for the SGD optimizer is:

$$W_{t+1} = W_t - \alpha \cdot \nabla J(W_t) \tag{10}$$

Here, W_t represents the weights at time step t, W_{t+1} is the updated weights at time step $t+1$, α is the learning rate, and $\nabla J(W_t)$ is the gradient of the loss function with respect to the weights W_t [44].

E. Gaussian Weighting Algorithm

To further enhance the accuracy of billboard detection, a Gaussian weighting was applied to the detected object's centre. During testing, the bounding boxes generated by each network were filtered using a Gaussian function centred at the detected object's centroid. This additional step aimed to refine the object localisation and improve the model's accuracy [36].

To implement the Gaussian Distribution algorithm, the pre-trained billboard model is loaded, and a confidence threshold of 50% is set for detection. For each test image, the algorithm calculates the coordinates of the image center and the center coordinates of the detected object. The distance x between the object center and the image center is computed as the euclidean distance. The object's weight, denoted as σ, is set to one-fourth of the image width $\frac{w}{4}$. The Gaussian weight is obtained by applying the Gaussian function with the distance, where the mean μ is set to 0 for the center. The equation for the Gaussian Distribution is given by [45] [46]:

$$Gaussian\ Distribution = \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{11}$$

The Gaussian weight is then multiplied by the confidence score of the detected object, resulting in the combined score. If the combined score surpasses the threshold of 50%, the algorithm identifies the billboard as the center one of interest and draws a green bounding box around it. Conversely, if the combined score is below the threshold, a red bounding box is drawn to indicate that the billboard is not located at the center. This can be observed in Figures 6 and 7. The Gaussian Distribution algorithm plays a crucial role in refining the results of billboard detection.

V. RESULT ANALYSIS

We conducted experiments using a dataset of 238 unseen images to analyse the generalisation capabilities of our model. The Mean Average Precision (mAP) is widely used as a metric to evaluate the object detection models [47], as it compares the ground-truth bounding box to the detected box and provides a score that reflects the accuracy of the model's detections. A higher score indicates more accurate detections [48].

TABLE II
MODEL COMPARISON RESULTS

Model	**Faster RCNN**	**CenterNet**	**YoloV5**	**YoloV8**
Step Size / Epoch	25,000	25,000	100	50
Training Accuracy (mAP.50)	59%	75.70%	64.50%	**87.40%**
Testing Accuracy (mAP.50)	54.30%	57.20%	52.60%	63.40%
With Gaussian - Testing Accuracy	65.43%	55.55%	80.24%	**82.71%**

Table II provides a comprehensive comparison of the performance of four different models: Faster RCNN, CenterNet, YOLOv5, and YOLOv8. The models were trained with different step sizes/epochs, and their performance was evaluated based on training accuracy and testing accuracy using mean Average Precision at 0.50 Intersection over Union (mAP@0.50). Furthermore, the testing accuracy was assessed through the application of Gaussian distribution weighting.

The training accuracy results for the models are as follows: Faster RCNN achieved a training accuracy of 59%, CenterNet achieved 75.70%, YOLOv5 achieved 64.50%, and YOLOv8 exhibited an impressive training accuracy of 87.40%. These results indicate the model's ability to learn and detect billboard features during the training process.

Using the testing dataset, the results are as follows: Faster RCNN achieved a testing accuracy of 54.30%, CenterNet achieved 57.20%, YOLOv5 achieved 52.60%, and YOLOv8 achieved 63.40%. These scores demonstrate the model's performance in accurately detecting billboards in unseen images. These results demonstrate that the YOLOv8 model obtains good performance compared with other models with respect to both training and testing.

In order to further enhance the accuracy of detecting billboards, we employed a Gaussian weighting technique specifically on the centre of the detected objects. This modification aimed to improve overall network performance by reducing incorrect classifications in a number of situations, namely those with multiple detections within a scene when wish to select the object that is closest to the centre of the image with a high confidence interval for the billboard class.

After applying the Gaussian weighting, we observed the following results in terms of testing accuracy for each model: Faster RCNN improved from 54.30% to 65.43%, CenterNet experienced a slight decrease from 57.20% to 55.55%, it performed relatively poorly during testing, indicating that although it had previously exhibited good generalisation during training by accurately detecting the object centres, it struggled to accurately identify the centres of billboard objects in the testing phase. YOLOv5 achieved a significant increase from 52.60% to 80.24%. Notably, YOLOv8 exhibited the most remarkable improvement in testing accuracy after the application of the Gaussian weighting technique. With an increase from 63.40% to an impressive 82.71%, it emerged as the top-performing model in terms of accurately detecting billboards for this dataset. These findings highlight the effectiveness of the Gaussian weighting technique in significantly enhancing the performance of the models.

Figures 6 and 7 illustrate examples of the application of the Gaussian distribution algorithm to detect the centre billboard in the case of multiple objects present in the scene and justifies the YOLOv8 testing accuracy increase. In Figure 6, the green bounding box highlights the selection of the center billboard, with a detection score of 80.60%. Significantly, even though there was an alternative billboard exhibiting a higher detection score of 94.66% (depicted by the adjacent red bounding box), its Gaussian weight of 15.06 was relatively lower, resulting in a

 ISBN: 978-1-68558-115-2

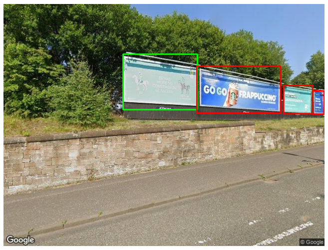

Fig. 6. Center Billboard Detection Results Using the Gaussian Distribution Algorithm.

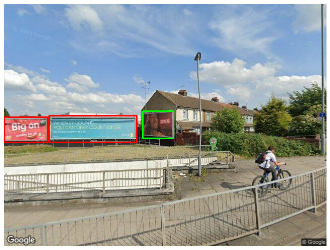

Fig. 7. Center Billboard Selection based on Combined Weights with Gaussian Algorithm.

combined weight that fell below the designated 50% threshold. Similarly, two additional billboards in the scene surpassed the 50% detection threshold with scores of 73.76% and 68.31%. However, due to their comparatively lower combined weights, they were not deemed as the center billboards of interest (indicated by the red bounding boxes).

Analogous findings were observed in Figure 7, reflecting the similar results depicted in Figure 6. The centre billboard was selected based on its confidence score of 84.89% and a Gaussian weight of 99.65%, outweighing another billboard that had a detection score of 92.45% but a lower Gaussian weight of 22.36% resulting in a combined weight of 20.67%, making it unsuitable for the center position. Furthermore, there was an additional billboard with a detection score of 70.00% and a Gaussian weight of 66.85%. However, the combination of this detection score and Gaussian weight fell below the threshold criteria of 50%, thus it was also not chosen as the centre billboard.

These results provide strong evidence supporting our hypothesis that applying a Gaussian weighting approach to objects based on their proximity to the centre of the image can greatly improve the accuracy of billboard detection. In summary, the comparison of the model's performance reveals that YOLOv8 consistently outperforms the other models in terms of both training and testing accuracy. However, it is important to consider other factors such as computational requirements, model complexity, and specific use case requirements when selecting the most suitable model for a given scenario.

VI. Conclusion

This research paper addresses the need for accurate billboard detection in advertising analytics. We have made notable contributions in several areas. Firstly, we developed a robust billboard detection system using advanced models like YOLOv8, YOLOv5, Faster-RCNN, and CenterNet. Challenges involved in this endeavor included finding optimal hyperparameters, mitigating over-fitting, and efficiently managing computational resources during the training process, all of which we adeptly addressed and resolved during the development of these networks. Hence, these models demonstrate high accuracy in detecting billboards in real-world scenarios.

Furthermore, we introduced an innovative approach by applying a Gaussian weighting technique to determine the most central billboards. This significantly improved the overall accuracy of the detection process, particularly in the case of YOLOv8, which achieved an impressive accuracy of 82.71% after the application of the Gaussian weighting. The combination of accurate detection models and the novel Gaussian weighting approach proves to be a promising direction for improving billboard detection in various domains. This advancement holds significant potential for applications such as urban planning, advertisement analysis, and traffic monitoring.

Future work could explore further refinements to optimise the proposed approach and extend it to real-time billboard detection systems, encompassing the task of verifying the relevance of displayed information. This would involve ensuring that billboards continuously present accurate and up-to-date content, addressing scenarios where some billboards may no longer convey valid information. Furthermore, exploring the billboard visibility based on environmental conditions, as well as the unique perspectives offered by different viewing angles, resulting in more effective outdoor advertising.

Acknowledgment

We wish to express our gratitude towards Digital Natives for providing the geo-coordinates for the dataset and funding the International PhD Scholarship at Ulster University.

References

[1] R. T. Wilson and J. Casper, "The role of location and visual saliency in capturing attention to outdoor advertising: How location attributes increase the likelihood for a driver to notice a billboard ad," *Journal of Advertising Research*, vol. 56, no. 3, pp. 259–273, 2016.

[2] S. Chavan, D. Kerr, S. Coleman, and H. Khader, "Billboard detection in the wild," pp. 57–64 Irish Machine Vision and Image Processing Conference 2021, Sep. 2021. [Online]. Available: https://iprcs.github.io/IMVIP.html [retrieved: Aug, 2023]

[3] K. Duan et al., "CenterNet: Keypoint triplets for object detection," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, pp. 6568–6577, 2019.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." [Online]. Available: http://image-net.org/challenges/LSVRC/2015/results [retrieved: Aug, 2023]

[5] G. Jocher et al., "ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation," Nov. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7347926 [retrieved: Aug, 2023]

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 779–788, 2016.

[7] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 to YOLOv8 and Beyond," pp. 1–31, 2023. [Online]. Available: http://arxiv.org/abs/2304.00501 [retrieved: Aug, 2023]

[8] K. Li and L. Cao, "A review of object detection techniques," *Proceedings - 2020 5th International Conference on Electromechanical Control Technology and Transportation, ICECTT 2020*, pp. 385–390, 2020.

[9] A. Dhillon and G. K. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence*, vol. 9, no. 2, pp. 85–112, 2020. [Online]. Available: https://doi.org/10.1007/s13748-019-00203-0 [retrieved: Aug, 2023]

[10] L. Alzubaidi et al., *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*. Springer International Publishing, 2021, vol. 8, no. 1. [Online]. Available: https://doi.org/10.1186/s40537-021-00444-8 [retrieved: Aug, 2023]

 ISBN: 978-1-68558-115-2

[11] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object Detection in 20 Years: A Survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[12] W. Liu et al., "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer International Publishing, 2016, pp. 21–37.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jun 2014, pp. 580–587. [Online]. Available: https://arxiv.org/abs/1311.2524 [retrieved: Aug, 2023]

[14] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015, pp. 1440–1448.

[15] T. Lin et al., "Feature Pyramid Networks for Object Detection," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944.

[16] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, 2020.

[17] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 5168–5177. [Online]. Available: https://arxiv.org/abs/1611.06612v3 [retrieved: Aug, 2023]

[18] K. He et al., "Spatial pyramid pooling in deep convolutional networks for visual recognition," *Computer Vision – ECCV 2014*, pp. 346–361, 2014.

[19] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable Part Models are Convolutional Neural Networks," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 437–446, sep 2014. [Online]. Available: https://arxiv.org/abs/1409.5403v2 [retrieved: Aug, 2023]

[20] Y. Li, Y. Chen, N. Wang, and Z. X. Zhang, "Scale-Aware Trident Networks for Object Detection," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, pp. 6053–6062, jan 2019. [Online]. Available: https://arxiv.org/abs/1901.01892v2 [retrieved: Aug, 2023]

[21] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, pp. 9626–9635, apr 2019. [Online]. Available: https://arxiv.org/abs/1904.01355v5 [retrieved: Aug, 2023]

[22] K. Chen et al., "Hybrid Task Cascade for Instance Segmentation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 4969–4978, jan 2019. [Online]. Available: https://arxiv.org/abs/1901.07518v2 [retrieved: Aug, 2023]

[23] X. Zhu et al., "Deformable DETR: Deformable Transformers for End-to-End Object Detection," oct 2020. [Online]. Available: https://arxiv.org/abs/2010.04159v4 [retrieved: Aug, 2023]

[24] A. Kirillov et al., "Segment Anything," 2023. [Online]. Available: http://arxiv.org/abs/2304.02643 [retrieved: Aug, 2023]

[25] R. F. Rahmat, Dennis, O. S. Sitompul, S. Purnamawati, and R. Budiarto, "Advertisement billboard detection and geotagging system with inductive transfer learning in deep convolutional neural network," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 17, no. 5, pp. 2659–2666, 2019.

[26] P. Liang et al., "Planar object tracking in the wild: A benchmark," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 651–658, 2018.

[27] A. Watve and S. Sural, "Detection of on-field advertisement billboards from soccer telecasts," *IET International Conference on Visual Information Engineering*, pp. 12–17, 2006.

[28] G. Cai, L. Chen, and J. Li, "Billboard advertising detection in sport TV," *Proceedings - 7th International Symposium on Signal Processing and Its Applications, ISSPA 2003*, vol. 1, pp. 537–540, 2003.

[29] H. Liu, L. Wang, W. Zhang, and W. Wang, "An illegal billboard advertisement detection framework based on machine learning," *ACM International Conference Proceeding Series*, pp. 159–164, 2019.

[30] V. Conotter and G. Boato, "Detecting photo manipulation on signs and billboards," *Proceedings - International Conference on Image Processing, ICIP*, no. 1, pp. 1741–1744, 2010.

[31] S. Anbukkarasi, V. E. Sathishkumar, C. R. Dhivyaa, and J. Cho, "Enhanced Feature Model based Hybrid Neural Network for Text Detection on Signboard, Billboard and News tickers," *IEEE Access*, vol. 11, no. April, pp. 41 524–41 534, 2023.

[32] K. Bochkarev and E. Smirnov, "Detecting advertising on building façades with computer vision," *Procedia Computer Science*, vol. 156, pp. 338–346, 2019. [Online]. Available: https://doi.org/10.1016/j.procs.2019.08.210 [retrieved: Aug, 2023]

[33] M. Hossari, S. Dev, M. Nicholson, K. McCabe, A. Nautiyal, C. Conran, J. Tang, W. Xu, and F. Pitié, "ADNet: A deep network for detecting adverts," *CEUR Workshop Proceedings*, vol. 2259, pp. 45–53, 2018.

[34] G. Liu, C. Wang, and Y. Hu, "RPN with the attention-based multi-scale method and the adaptive non-maximum suppression for billboard detection," *2018 IEEE 4th International Conference on Computer and Communications, ICCC 2018*, pp. 1541–1545, 2018.

[35] Á. Morera, Á. Sánchez, A. B. Moreno, Á. D. Sappa, and J. F. Vélez, "Ssd vs. Yolo for detection of outdoor urban advertising panels under multiple variabilities," *Sensors (Switzerland)*, vol. 20, no. 16, pp. 1–23, 2020.

[36] L. Hou, K. Lu, X. Yang, Y. Li, and J. Xue, "G-Rep: Gaussian Representation for Arbitrary-Oriented Object Detection," *Remote Sensing*, vol. 15, no. 3, pp. 1–21, 2023.

[37] "The neuron intelligent connections: Programmatic exchange for dooh advertising." [Online]. Available: https://theneuron.com/ [retrieved: Sept, 2023]

[38] I. O. Muraina, "Ideal Dataset Splitting Ratios in Machine Learning Algorithms: General Concerns for Data Scientists and Data Analysts," *7th International Mardin Artuklu Scientific Researches Conference*, no. February, pp. 496–504, 2022.

[39] Y. Chen et al., "Domain Adaptive Faster R-CNN for Object Detection in the Wild," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3339–3348, 2018.

[40] Y. Guo and X. Lu, "ST-CenterNet: Small Target Detection Algorithm with Adaptive Data Enhancement," *Entropy (Basel, Switzerland)*, vol. 25, no. 3, 2023. [Online]. Available: https://doi.org/10.3390/e25030509 [retrieved: Aug, 2023]

[41] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," 2023. [Online]. Available: http://arxiv.org/abs/2305.09972 [retrieved: Aug, 2023]

[42] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: challenges, architectural successors, datasets and applications," *Multimedia Tools and Applications*, vol. 82, no. 6, pp. 9243–9275, 2023.

[43] J. Doherty, B. Gardiner, E. Kerr, N. Siddique, and S. Manvi, "Comparative study of activation functions and their impact on the yolov5 object detection model," *International conference on pattern recognition and artificial intelligence, ICPRAI*, pp. 40–52, Jun. 2022.

[44] S. Ruder, "An overview of gradient descent optimization algorithms," pp. 1–14, 2016. [Online]. Available: http://arxiv.org/abs/1609.04747 [retrieved: Aug, 2023]

[45] T. Beckers, "An Introduction to Gaussian Process Models," *arXiv preprint arXiv:2102.05497*, 2021.

[46] X. Zhang, "Gaussian distribution," *Encyclopedia of Machine Learning*, pp. 425–428, 2010.

[47] R. Padilla et al., "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics (Switzerland)*, vol. 10, no. 3, pp. 1–28, 2021.

[48] J. Revaud, J. Almazán, R. S. Rezende, and C. R. d. Souza, "Learning with average precision: Training image retrieval with a listwise loss," pp. 5107–5116, 2019.