International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 15, No.1 (May-24)

http://dx.doi.org/10.12785/ijcds/1501119

Human Detection in Clear and Hazy Weather Based on Transfer Learning With Improved INRIA Dataset Annotation

Yassine Bouafia [1], Larbi Guezouli [2] and Hicham Lakhlef [3]

[1] *Department of Computer Science, University of Batna 2, LaSTIC Laboratory, Batna, Algeria*
[2] *LEREESI Laboratory, HNS-RE2SD, Batna ,Algeria*
[3] *Sorbonne University, University of Technology of Compiègne, Compiègne, France*

Received 13 Feb. 2024, Revised 19 Mar. 2024, Accepted 30 Mar. 2024, Published 1 May. 2024

Abstract: Human detection plays a pivotal role in many vision-based applications. Effectively detecting humans across diverse environments and situations significantly contributes to enhancing human safety. However, this effectiveness encounters challenges, particularly in hazy conditions that reduce visibility and blur images, thereby impacting the accuracy of existing detection algorithms. Additionally, the quality of dataset annotations significantly affects the accuracy of these systems. Poor annotations lead to insufficient training of detection models, resulting in higher error rates and reduced efficacy in real-world scenarios. To tackle these challenges, we've introduced new, more precise annotations for the INRIA dataset. These enhancements overcome limitations within the dataset, particularly instances where numerous individuals in images lacked proper labeling. This augmentation aims to improve training robust detection models and provide a more accurate evaluation of the model's performance. Our experiments have yielded notable improvements, showcasing a 20.37% increase in Average Precision and a substantial 68.19% reduction in False Negatives. Moreover, we've developed a deep-learning model for human detection, leveraging transfer learning to fine-tune the YOLOv4 model. Experimental results demonstrate that our proposed model accurately detects pedestrians under various weather conditions, including both clear and hazy scenarios. It achieves high average precision and F-Scores while maintaining efficient real-time operation at 55.4 FPS. These advancements significantly enhance the reliability and applicability of human detection systems.

Keywords: Human Detection, CNN, Deep Learning, YOLOv4, Transfer Learning, INRIA Dataset

1. Introduction

Object detection has attracted great interest in recent years. Human detection is a sub-problem that remains a major research topic due to its diverse applications, such as surveillance cameras [1], unmanned aerial vehicles (UAV) [2], biometric systems [3] and tumor detection [4]. Human detection is a particularly important and essential task in any intelligent video surveillance system. Indeed, detecting humans in various environments can help to prevent crime, theft, avoid incidents and improve the driving safety of autonomous vehicles. However, there are still several factors that make human detection challenging. For instance, the human body is flexible, giving rise to a wide variety of poses. People also wear clothes of different colors and textures, which creates an additional complication. In addition, different weather conditions like rain, snow, and haze make the operation more difficult.

In the past, object detection required a manual process of extracting features using a sliding window and inputting them into a classifier. Common representative features included Histograms of Oriented Gradients (HOG) and deformable part models (DPM), Haar and SIFT. Classification techniques were typically categorized into supervised and unsupervised methods. Supervised approaches commonly employed the Support Vector Machine (SVM) or Perceptron, while unsupervised methods typically utilized K-means and Mean-shift. Following the deep learning revolution, the use of detection methods based on deep learning architectures has also grown. Initially, the application of deep learning approaches focused mainly on two-stage detection algorithms. These methods first extract candidate regions. Then, they use classification networks to classify the extracted candidate regions. The most popular variants of the two-stage detectors are the R-CNN models [5], [6], [7]. On the other hand, one-step detection methods skip the region-of-interest selection process and, instead, employ bounding box regression to perform detection and recognition on the same time, ensuring a simple, consistent flow from start to finish. The most familiar single-shot models are the SSD [8] and the YOLO family [9], [10], [11], [12], [13], [14], [15].

However, the majority of human detection algorithms

E-mail address: y.bouafia@univ-batna2.dz, larbi.guezouli@hns-re2sd.dz, hicham.lakhlef@utc.fr

have primarily been evaluated under clear weather conditions. These methods often struggle to detect humans in low-light scenarios. The presence of haze poses a significant challenge due to reduced visibility, object blurring, and difficulty in distinguishing humans from the background. Previous research efforts have introduced dehazing methods aimed at restoring and enhancing image contrast [16], [17], [18], [19]. But, these techniques are typically designed for daylight scenarios with uniform light distribution. Consequently, they prove less effective in low-light conditions where haze is more prevalent.

Furthermore, data is a critical component of any artificial intelligence model and, fundamentally, the main cause for the massive growth in the popularity of machine learning that we are witnessing today. The human detection community has widely used the popular INRIA person dataset to train and evaluate detectors. Nevertheless, this dataset has some limitations, as many individuals appearing in the images are not labelled, which affects the learning ability of the model as well as the results reported during the evaluation phase.

Therefore, to address these issues, our research is motivated by two main objectives aimed at improving human detection in challenging environments. Firstly, to address the limitations inherent in existing human detection models that struggle under hazy weather conditions. By leveraging a one-step deep learning model using transfer learning technique to develop a more efficient real-time model capable of detecting humans in clear and hazy conditions. Secondly, to address the issue of incomplete and inaccurate labeling in the INRIA person dataset. By creating new improved annotations of the dataset. The use of these new annotations for training will lead to better performance, as well as more accurate evaluations of the model during the test phase. The main contributions of this work are as follows:

- The introduction of improved labelling for the INRIA dataset, addressing its existing limitations and enhancing the dataset's utility for more effective training and evaluation. Our experiments demonstrate significant performance enhancements, including a 20.37% increase in Average Precision and a notable reduction in False Negatives by 68.19%.
- Development of a one-stage deep learning model capable of effectively detecting humans, even in challenging weather conditions such as haze. The use of a one-stage detector makes it faster and more energy-efficient than several widely used Deep Convolutional Neural Network (DCNN) models. Its practicality and adaptability make it particularly suitable for real-life applications, ensuring reliable performance across diverse environmental conditions.
- Exploration of the impact of input image size on model performance, establishing a correlation between the model's efficiency, the dataset's average dimensions, and the chosen input image size.

The remainder of the paper is organized as follows. Section 2 provides an overview of related works. Next, in the Section 3 we describe the limitations of the current INRIA labelling and the advantage of our new labelling. Also, we describe the implementation of our model using transfer learning. In Section 4, we present all the experiments we have done and also the discussion of obtained results. Finally, the conclusion is given in Section 5.

2. Related Works

In recent years, object detection methodologies have evolved significantly. Initially, efforts were primarily centered around the sliding window search and handcrafted features like Histograms of Oriented Gradients (HOG) [20], integral channel features (ICF) [21], aggregated channel features (ACF) [22], and deformable part models (DPM) [23]. Classifiers such as support vector machines (SVMs) [24] and adaptive boosting (AdaBoost) [25] were employed with these features for object recognition. Mihçioğlu et al. [26] utilized HOG features for pedestrian detection, marking considerable advancements in the field. Kumar et al. [27] enhanced pedestrian detection accuracy by characterizing pedestrian shape and texture features using Histogram of Significant Gradients (HSG) and Non Redundant Uniform Local Binary Pattern (NRULBP), combined with an SVM classifier. However, due to the complexity and variability of real-world scenes, these handcrafted feature-based methods result in models with limited generalizability and poor robustness, failing to satisfy realistic requirements [28].

With the increase in computing power and the number of image datasets, deep learning has quickly changed the world of artificial intelligence. Convolutional neural networks (CNNs), in particular, have revolutionized the field of image recognition, demonstrating their efficiency and precision in processing visual data. Deep learning methods for object detection generally can be classified into two main families.

Two-stage detectors: These models operate in two stages. Initially, the model will suggest a set of regions of interest using techniques like selective search or regional proposal networks. In the subsequent stage, the model performs classification on these regions and refines the location predictions. While two-stage detectors often achieve higher accuracy, they tend to be slower compared to one-stage detectors. Some popular models of two-stage object detectors include R-CNN [5], Fast-RCNN[6], Faster-RCNN[7], Mask-RCNN[29] and Cascade RCNN [30].

One-stage detectors: These models approach object detection as a regression task, managing both object classification and bounding box regression directly, without depending on previously generated region proposals. One-stage detectors exhibit high inference speed and lower computational requirements but typically offer relatively lower accuracy compared to region-based methods. Notable examples of single-shot models include SSD [8], Reti-

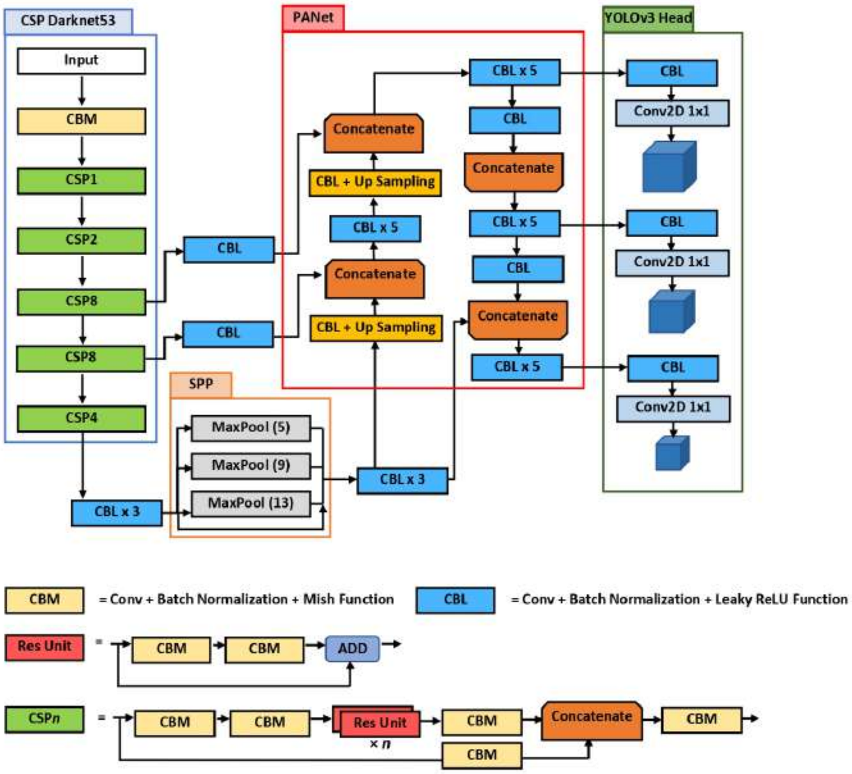


Figure 1. YOLOv4 structure diagram

naNet [31] and the most popular model is YOLO (You Only Look Once) [32], which yields a good performance. However it suffers from the challenge of overwhelming negative samples. A series of improved YOLO versions has been developed [9], [10], [11], [12], [13], [14], [15], consolidating YOLO's position as one of the best one-stage object detectors. The most recent versions are YOLOv7 [14] and YOLOv8 [15]. Furthermore, lightweight variants of the model, known as YOLO-tiny or YOLO-small, have been created to address the problem of resource limitations.

These algorithms have both strengths and weaknesses. As a result, researchers are now focusing on the challenges faced in human detection task. Detection efficiency is affected by issues such as the sensitivity to both internal and external environmental conditions, as well as the complication of occlusions in complex environments. In the field of autonomous driving, it is essential that detection speeds align with real-time operational requirements. It is equally crucial to maintain detection accuracy in adverse weather conditions. Therefore, in order to ensure better adaptation to the pedestrian detection task, many researchers have proposed improved methods. Wu et al. [33] introduced a new Self-Mimic Learning approach to improve the detection performance of small pedestrians based on Faster-RCNN. Kyrkou [34] proposed a one-stage pedestrian detection approach named YOLOpeds, which integrates dense connections between layers and the fusion of multi-scale features to enhance representational capacity while simultaneously reducing the number of operations and parameters. Li et al. [35] proposed a lightweight pedestrian detection approach which is based on the YOLOv5 architecture. They apply the Ghost modules to minimize computational costs during feature extraction process. They also integrate the Global Attention Mechanism (GAM) module to improve feature extraction accuracy. Based on RetinaNet, Huang et al. [36] used a multi-branch structure with a double-pooling attention mechanism to extend the network and enhance the cross-channel feature information correlation and improve model detection accuracy. But these methods often struggle to detect people in low-light scenarios. Particularly in the presence of haze, the reduced visibility, blurring of objects and difficulty in distinguishing people from the background pose a significant problem. Earlier research has introduced dehazing methods to restore and improve image contrast [16], [17], [18], [19]. However, these techniques are generally designed for daylight scenarios with uniform light distribution. As a result, in low-light conditions where haze is more frequent, they are less effective.

A. YOLOv4

Known for its remarkable speed and accuracy in object detection, YOLOv4 [11] forms the foundation of our model. Through the incorporation of mosaic data enhancement in data processing and the optimization of backbone, network training, activation, and loss functions, YOLOv4 stands out as a robust real-time object detector. The YOLOv4 architecture harnesses the Cross Stage Partial Darknet53 (CSPDarknet-53) as its fundamental backbone network to process and extract image features. Complementing this, the Spatial Pyramid Pooling (SPP) block is integrated into CSPDarknet-53, effectively expanding the receptive field and isolating crucial contextual features. Unlike YOLOv3, which utilizes Feature Pyramid Networks (FPN) for object detection, YOLOv4 employs the Path Aggregation Network (PANet) to aggregate parameters across various detector levels. Notably, YOLOv4 maintains the original YOLOv3 network architecture for the detector head. Figure 1 illustrates the architecture of YOLOv4.

B. Transfer Learning

We humans have a natural skill in transferring knowledge from one ”task A” to another ”task B”. The knowledge we acquire when solving one task, we use in the same way to solve related tasks. For example, if you know how to drive a motorcycle, you use this knowledge to learn how to drive a car. In transfer learning, we attempt to use what we have learned in one task to increase generalization in another. So, we transfer the weights that one network learns in ”task A” to another network to use them to learn how to solve a different ”task B”, as illustrated in Figure 2. Transfer learning enables the pre-trained YOLOv4 model to adapt to our specific task. This process involves fine-tuning the model with our dataset, which comprises images in clear and hazy weather. The pre-learned features from the YOLOv4 model are refined using our dataset, leading to improved detection accuracy.

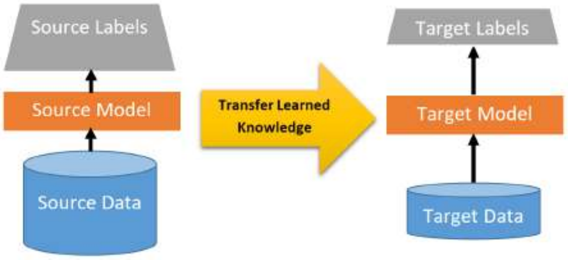


Figure 2. Transfer learning approach

Transfer learning is suitable for creating efficient deep-learning models for several reasons:

- Reduced Training Time: Transfer learning allows us to leverage the pre-trained model’s knowledge and use it as a starting point for the new task. This means we can train the new model with fewer iterations and epochs, which saves time and computational resources.
- Improved Generalization: The pre-trained model has already learned a lot of features that are useful for many different tasks. By using this knowledge as a starting point, we can improve the generalization of the new model, allowing it to perform better on unseen data.
- Better Performance: Pre-trained models are usually trained on large and diverse datasets, allowing them to capture more complex patterns and relationships in the data. By using this pre-trained knowledge, we can improve the performance of the new model, even with a smaller dataset.
- Fewer Data Requirements: Deep learning models require a lot of data to learn effectively. By using a pre-trained model as a starting point, we can reduce the amount of data required to train the new model. This is especially important for domains where collecting large amounts of labeled data is difficult or expensive.
- Flexibility: Transfer learning allows us to use pre-trained models for different tasks, without having to train new models from scratch. This makes deep learning more accessible to a wider range of applications and reduces the time and resources required to develop new models.

3. Methodology

A. Improving The Inria Dataset Annotations

In the pedestrian detection community, the INRIA person dataset is a popular resource for training detectors and communicating results. Introduced by Dalal and Triggs in 2005 [24], this dataset comprises 614 training images, including 2416 pedestrians, and 288 testing images, encompassing 1126 pedestrians. However, notable limitations exist due to the absence of labels for many individuals appearing in the images, as illustrated in Figure.3-a.

Labeling a dataset serves multiple purposes. Firstly, it allows for the extraction of positive samples crucial for detector training. Secondly, it enables the use of test set annotations for evaluation, facilitating the identification of correct detections. During the learning phase, the presence of unlabeled individuals is counted as false predictions, impacting the model’s learning capacity. Consequently, the resulting model overlooks many individuals in the image, leading to a high False Negative (FN) rate. Moreover, during evaluation, each detection of an unlabeled person is recorded as a False Positive (FP) instead of a True Positive (TP), providing inaccurate insights into the model’s performance.

To address this issue, we propose a novel annotation approach for the INRIA dataset, manually labeling all pedestrians visible within the images (as depicted in Figure.3-b), denoted as **INRIA-N**. Enhanced labeling is anticipated to significantly improve model learning, consequently enhancing overall performance and reducing the

FN rate. Additionally, refining person labeling within the test set promises more precise evaluations of the model's performance.

B. Fine-tuning And Training

In this section, we describe the process of fine-tuning and training YOLOv4 for human detection. The diagram presented in Figure 4 serves as a visual guide, outlining the sequential steps involved in constructing a our model through the application of transfer learning technique. Initially, after creating the new annotations for the INRIA dataset, we prepared our dataset by converting the annotations to the YOLO Darknet format. Utilizing Google Colab [37], we set up the Darknet environment, the core of YOLOv4. After that, we need to fine-tune the YOLOv4 model to perform the human detection task. Thus, the configuration of default hyperparameters changed as shown in Table I. We initiate the training process with pre-trained weights from the MS-COCO dataset, providing a strong foundation of fundamental knowledge. We chose to make all the weights of the model trainable, rather than freezing some layers as is common in transfer learning. This strategy overcomes the limitations of transfer learning, where pre-trained layers may not align perfectly with new data, leading to sub-optimal feature extraction. By making all weights trainable, we ensure that each layer of the model doesn't just apply pre-learned patterns, but actively learns and adapts to the characteristics of our dataset. This enables more efficient feature extraction, improving overall model performance. Finally, we trained the fine-tuned YOLOv4 on Google Colab with a Tesla K80 GPU. Figure 4 shows the transfer learning process.

TABLE I. Hyperparameters configuration

Hyperparameters	Values	Explication
Number of classes	1	
Max batches	6000	classes × 2000. But not less than the number of training images and not less than 6000 [38].
Filters size	18	(classes + 5) × 3, where 5 represents 4 bounding boxes coordinates +1 object prediction value and 3 the number of masks [38].
Image size	416×416	
Batch size	64	This means that for each training step, the system will use 64 images.
Subdivisions size	16	To reduce GPU VRAM consumption, batch sizes will be divided by subdivision sizes

4. Experiments And Results

A. Datasets

We conducted experiments on three datasets INRIA-N, HazePerson and Caltech.

1) INRIA-N

The INRIA person dataset was introduced by Dalal and Triggs in 2005 [24]. It provides 614 images for training and 288 for testing. Since many persons are not labelled in the original INRIA test dataset, we used INRIA-N dataset in the evaluation phase to get more accurate and truthful results about the model's performance. INRIA-N contains the same images as the original INRIA and uses our proposed annotations.

2) HazePerson

HazePerson was created by Guofa Li et al. [39] to tackle the problem of pedestrian detection in hazy weather. This dataset contains images of pedestrians in hazy weather with bounding box annotations. It provides 1052 images for training and 143 images for testing.

3) Caltech

The Caltech Pedestrian dataset comprises 11 sets, with the initial six sets (set00 – set05) designated for training, and the subsequent five sets (set06–set10) allocated for testing. Each set is composed of video footage captured from a vehicle navigating through normal urban traffic. For training purposes, one frame is extracted every 30 frames from the first six sets (set00 – set05), and similarly, one frame every 30 frames is extracted from the last five sets (set06–set10) for testing. The dataset includes approximately 250,000 frames, with around 350,000 annotated bounding boxes (BBs). The frames are originally sized at 480×640.

B. Evaluation Metrics

To evaluate the models' performance on the INRIA-N and HazePerson test datasets, here are the metrics we used where TP = True Positive, TN = True Negative, FP = False Positive, and FN = False Negative.

Precision: is the ratio of correctly predicted positive observations to the total predicted positive observations.

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

Recall: is the ratio of correctly predicted positive observations to all observations in positive class.

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

F-Score: is the harmonic mean of the model's precision and recall. The formula for the F-Score is:

$$F\text{-}score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \tag{3}$$

Average precision (AP): a popular measure for evaluating the accuracy of object detectors. AP calculates the average precision value for recall values between 0 and 1.

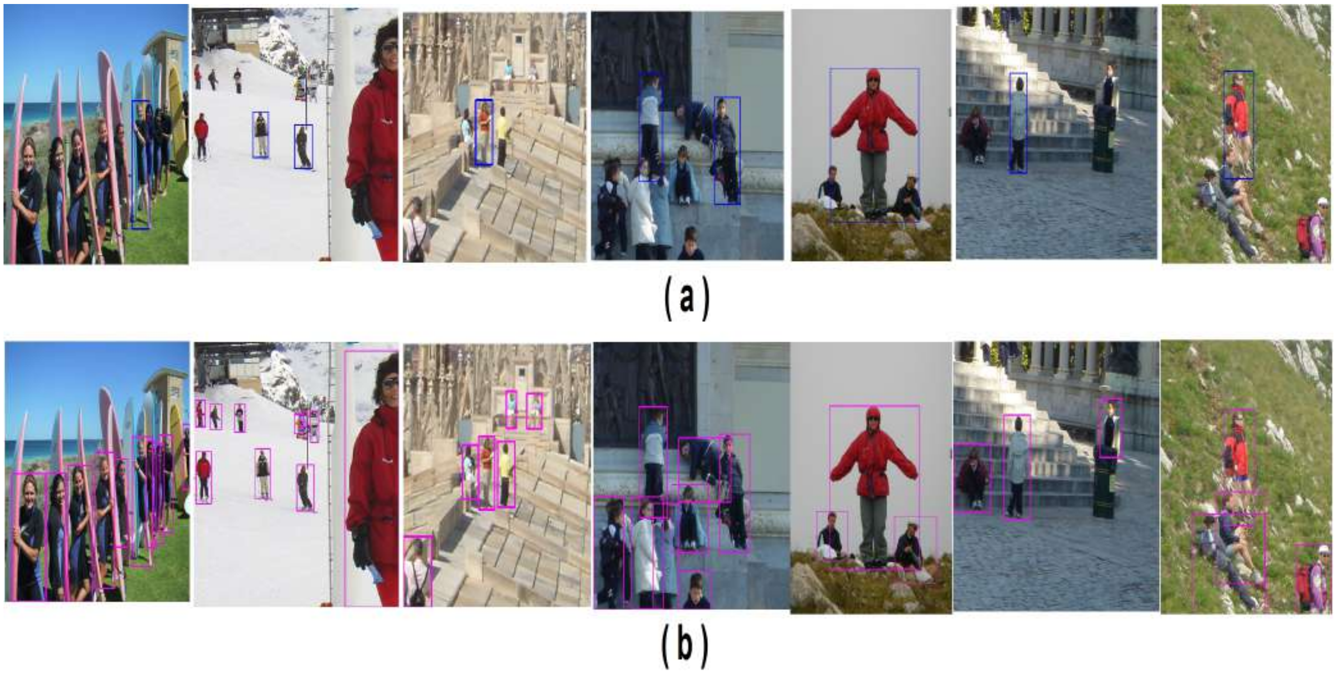


Figure 3. a - Original INRIA annotations, b – Proposed annotations

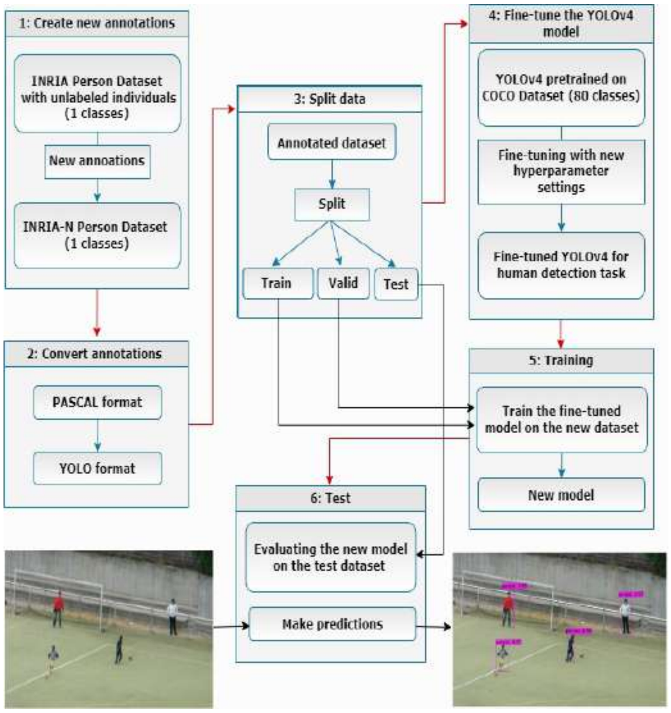

1: Create new annotations
INRIA Person Dataset with unlabeled individuals (1 classes)
New annoations
INRIA-N Person Dataset (1 classes)
2: Convert annotations
PASCAL format
YOLO format
3: Split data
Annotated dataset
Split
Train
Valid
Test
4: Fine-tune the YOLOv4 model
YOLOv4 pretrained on COCO Dataset (80 classes)
Fine-tuning with new hyperparameter settings
Fine-tuned YOLOv4 for human detection task
5: Training
Train the fine-tuned model on the new dataset
New model
6: Test
Evaluating the new model on the test dataset
Make predictions

Figure 4. Diagram showing the steps to build a fine-tuned YOLOv4 for human detection using the transfer learning technique

TABLE II. Evaluation settings for Caltech Pedestrian dataset

Subset	Pedestrian height (pixels)	Occlusion
Reasonable	≥ 50	$\geq 65\%$
All	≥ 20	$\geq 20\%$
Scale-large	≥ 100	-
Scale-near	≥ 80	-
Scale-medium	30 - 80	-
Scale-far	20 - 30	-

Log-average miss rate (LAMR): For the Caltech dataset, detection efficiency is measured using LAMR, determined by averaging the miss rate across nine FPPI (False Positive Per-Image) rates that are evenly distributed on a logarithmic scale from 10^{-2} to 10^{0}. This approach condenses the entire miss-rate versus FPPI curve into a single, easily comparable figure, where a smaller value indicates superior detection capability. The evaluation of detection performance on the Caltech dataset incorporates various criteria based on the height and the visible portion of the bounding boxes, as detailed in the parameters outlined in Table II.

C. *Effect of Using The New Annotations*

This section evaluates the improvement in detection performance using the new annotations. We conducted an experiment where we trained the fine-tuned YOLOv4 model with both the original and the newly proposed INRIA annotations. Due to the original INRIA test dataset's lack of comprehensive labelling (many persons are not labelled), we relied on our new annotations for a more accurate assessment of model performance. The results detailed in Table III, compare the performance of two models: O-YOLOv4 (trained with original annotations) and N-YOLOv4 (trained with new annotations). The findings indicate a substantial improvement in AP by 20.37% when using the new annotations. Moreover, the original annotations led to a high False Negative rate, missing many people in the images. In contrast, the new annotations significantly reduced FNs from 371 to 118, a decrease of 68.19%. Figure 5 illustrates these differences. For instance, O-YOLOv4 missed detecting some individuals in the images, while N-YOLOv4 demonstrated superior performance, even detecting smaller people at the top of the image as shown in Figure 5 (a.2) or partially obscured individuals, such as those at a distance or behind obstacles like cars as shown in Figure 5 (a.4).

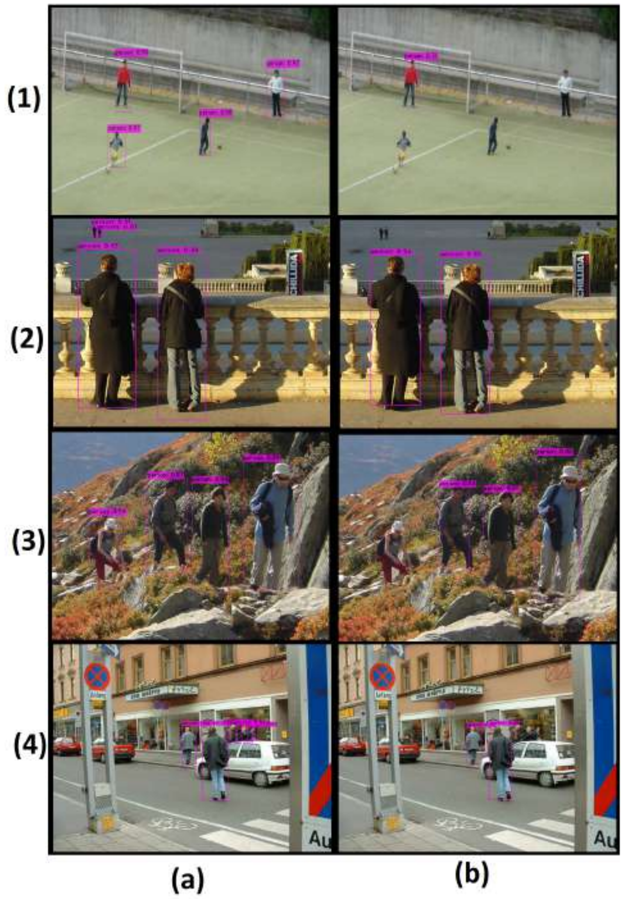


Figure 5. Detection results. (a) N-YOLOv4: trained using proposed annotations. (b) O-YOLOv4: trained using original INRIA annotations.

D. Comparison With Other Methods on INRIA-N Test Set

In our comparative analysis, we evaluated N-YOLOv4 against various YOLO models on the INRIA-N dataset. Before the evaluation, we trained all models on the INRIA-N training set with default settings to enhance their performance on this database and maintain consistency. It can be seen from Table III that N-YOLOv4 shows the best performance with an AP of 93.55%. N-YOLOv4, surpasses YOLOv3 by 12.03%, YOLOv6 by 5.89%, and YOLOv7 by 2.28%. Notably, it surpassed YOLOv8, the latest YOLO version, by 34.76%. N-YOLOv4's F1 score of 0.90 demonstrates its balanced precision and recall, outperforming other models such as YOLOv3, YOLOv6 and YOLOv8. Even compared with YOLOv7, which scores 0.93, N-YOLOv4 still achieves competitive results. The improved AP is complemented by a competitive F1-score and a speed of 55.4 FPS. These results make N-YOLOv4 a highly effective model for real-time human detection

E. Human Detection in Hazy Weather

An effective human detector could perform well in various environments and weather conditions. To further evaluate the flexibility of our human detector, particularly in difficult conditions, such as hazy weather where the appearance of pedestrians is not clear, we carried out tests using the HazePerson dataset. The performance of N-YOLOv4 on this dataset is shown in Table IV. Although N-YOLOv4 has not been trained on images in a similar situation, it still shows solid performance with an AP of 82.2% and an F1-score of 0.83. These results highlight the strong generalization capabilities of the model, which adapts effectively to new visually demanding scenarios.

F. The Effect of The Input Size

In the first experiments, we used an input size of 416x416 to test the model. In this experiment, we tried several input sizes during the evaluation to explore the effect of input size on model performance. We should notice here that the value of the input image size in YOLO should be a multiple of 32. The results, detailed in Table V, are also visually represented in a bar chart (Figure 6)

To show the impact of the size of images on the accuracy, we have increased the size of images to 480x480, 544x544, and 640x640, and then we decreased the size to 352x352, 288x288, and 244x244. As shown in Figure 6. We observed that increasing the input size (up to 640x640) enhances performance on the INRIA-N dataset, while decreasing it (down to 244x244) yields better results on the HazePerson dataset To find the reason for this difference in performance from one dataset to another and the effect of the input size on these results, we calculated the average of the height and width of the images in both datasets. The result was 631x631 for the INRIA test dataset and 236x236 for the HazePerson test dataset. This comparison revealed a correlation between the model's performance and the proximity of the input image size to the average image size in the datasets. The closer the input size matched the average dimensions (631x631 for INRIA and 236x236 for HazePerson), the better the model performed. Conversely, a significant deviation from these dimensions resulted in reduced effectiveness.

G. Merging INRIA and HazePerson Datasets

In the preceding experiment, our proposed model demonstrated strong performance on the HazePerson dataset, despite not being trained on any of its images. For this experiment, we amalgamated images from both the

TABLE III. Detection results on the INRIA-N dataset

Model	AP	TP	FP	FN	Precision	recall	F1-score	FPS
O-YOLOv4	73.18	503	4	371	0.99	0.58	0.73	-
YOLOv3	81.52	657	44	217	0.94	0.75	0.83	63.8
YOLOv6	87.66	770	48	104	0.94	0.88	0.91	33.35
YOLOv7	91.27	803	57	71	0.93	0.92	0.93	66.66
YOLOv8	58.79	487	2	387	0.99	0.56	0.71	29.33
N-YOLOv4	93.55	756	54	118	0.93	0.86	0.90	55.40

TABLE IV. The performance of our model on the HazePerson dataset

Model	Input size	AP	TP	FP	FN	Precision	recall	F1-score
N-YOLOv4	(416×416)	82.2	193	25	52	0.89	0.79	0.83

TABLE V. Detection results using different input size

Input size	INRIA-N	HazePerson
640×640	95.71	65.15
608×608	95.03	68.35
544×544	95.28	72.76
480×480	94.43	78.67
416×416	93.55	82.2
352×352	92.44	87.47
288×288	89.23	92.68
224×224	84.70	94.42

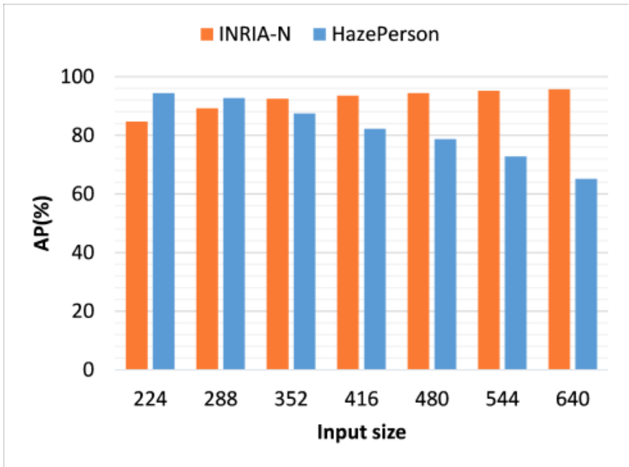


Figure 6. Summary of detection results using different input sizes.

INRIA and HazePerson datasets to create a novel training dataset. This combined dataset was utilized to fine-tune the YOLOv4 model, resulting in the creation of HaIN-YOLOv4. Table VI and Table VII outline the performance metrics of HaIN-YOLOv4 on the INRIA-N and HazePerson datasets across various input sizes.

We can see from these tables that HaIN-YOLOv4 (224×224) achieved better results on the HazePerson dataset with AP equal to 98.01%, which represents an improvement of 3.59% compared to N-YOLOv4 (240×240). Also, HaIN-YOLOv4 (640×640) maintained a good performance on the INRIA-N dataset by obtaining an AP equal to 95.21% and an F1-score equal to 0.92.

H. Comparison With Other Methods on The HazePerson Test Set

Our comparative analysis includes the deep learning approaches developed by Guofa Li et al. [39] for pedestrian detection in hazy conditions: Simple-Yolo, VggPrioriBoxes-Yolo, and MNPrioriBoxesYolo. This comparison, as detailed in Table VIII, is particularly relevant because they used the same HazePerson dataset for the training and evaluation of their models. This commonality in dataset usage provides a robust and fair basis for comparing the performance of our models against theirs. The comparison, presented in Table VIII, shows that N-YOLOv4 and HaIN-YOLOv4 outperform all other models on all metrics. Among all the methods from other studies, we note that MNPrioriBoxes-Yolo gives the best results. If we compare it with HaIN-YOLOv4, which uses the same input size, we note that HaIN-YOLOv4 achieved an AP equals to 98.01% and F-Score of 90%, surpassing MNPrioriBoxes-Yolo by 11.41% in AP and 3% in F-Score. Figure 7 illustrates the detection results under challenging conditions like haze, unclear vision, and poor lighting, where we can see that the model was able to detect the humans in the images with high efficiency.

I. Evaluation on Caltech Pedestrian Dataset

We have further evaluated our model on the Caltech Pedestrian dataset, a prominent benchmark in pedestrian detection. Firstly, we extract one frame every 30 frames from the first six sets (set00 – set05) of the Caltech dataset and have used it to train the proposed fine-tuned model. We named the resulting model Cal-YOLOv4. In our comparison, Cal-YOLOv4 was compared against some state-of-the-art models on the Caltech test set, including ACF+SDt, SCF+AlexNet, TA-CNN, Checkerboards, ACF++, DeepParts, CompACT-Deep, MS-CNN, RPN+BF. The compari-

TABLE VI. Detection results of N-YOLOv4 and HaIN-YOLOv4 on INRIA-N test dataset using different input size

Model	Input size	INRIA-N dataset						
		AP	TP	FP	FN	precision	recall	F1-score
N-YOLOv4	416×416	93.55	756	54	118	0.93	0.86	0.90
	640×640	95.71	769	50	105	0.94	0.88	0.91
	224×224	84.70	652	30	222	0.96	0.75	0.84
HaIN-YOLOv4	416×416	92.60	747	38	127	0.95	0.85	0.90
	640×640	95.21	787	45	87	0.95	0.90	0.92
	224×224	83.03	665	50	209	0.93	0.76	0.84

TABLE VII. Detection results of N-YOLOv4 and HaIN-YOLOv4 on HazePerson test dataset using different input size

Model	Input size	HazePerson dataset						
		AP	TP	FP	FN	precision	recall	F1-score
N-YOLOv4	416×416	82.20	193	25	52	0.89	0.79	0.83
	640×640	65.15	136	29	109	0.82	0.56	0.66
	224×224	94.42	222	23	23	0.91	0.91	0.91
HaIN-YOLOv4	416×416	96.86	240	64	5	0.79	0.98	0.87
	640×640	92.86	232	65	13	0.78	0.95	0.86
	224×224	98.01	240	47	5	0.84	0.98	0.90

TABLE VIII. Comparison of detection results of each model on the HazePerson dataset

Model	Input size	Precision	Recall	F1-score	AP
Simple-Yolo	224×224	0.77	0.70	0.73	62.7
VggPrioriBoxes-Yolo	224×224	0.85	0.84	0.85	80.8
MNPrioriBoxes-Yolo	224×224	0.88	0.89	0.87	86.6
N-YOLOv4	224×224	0.91	0.91	0.91	94.42
HaIN-YOLOv4	224×224	0.84	0.98	0.90	98.01

TABLE IX. Log-Average Miss Rate of detection results for each model on the Caltech dataset

Model	Reasonable (%)	All (%)	Large (%)	Near (%)	Medium (%)	Far (%)
ACF+SDt	37.34	77.01	14.19	20.97	69.55	100
SCF+AlexNet	23.32	70.33	7.01	10.61	62.34	100
TA-CNN	20.86	71.22	7.00	7.96	63.62	100
Checkerboards	18.47	68.75	3.90	6.07	59.42	100
ACF++	11.71	69.07	3.56	5.10	60.46	100
DeepParts	11.89	64.78	4.37	4.78	56.42	100
CompACT-Deep	11.75	64.44	2.64	3.99	53.23	100
MS-CNN	9.95	60.95	1.99	2.60	49.13	97.23
RPN+BF	9.85	64.66	1.18	2.26	53.93	100
Cal-YOLOv4	9.94	54.23	0.61	1.42	33.69	73.20

son, detailed in Table IX and Figure 8, shows Cal-YOLOv4 achieving a LAMR of 9.94% on the “Reasonable” setting, which is better than most existing methods. While RPN+BF slightly outperforms our model in this setting, with a LAMR of 9.85%. Cal-YOLOv4 excels in other settings like ”All,” ”Medium,” ”Near,” ”Large,” and ”Far,” demonstrating its remarkable effectiveness across various evaluation criteria.

5. Conclusion

In this study, we tackled the limitations in labeling of the INRIA person dataset by introducing improved annotations. These enhancements significantly boosted model performance and evaluation accuracy. Additionally, we developed a one-stage deep learning model for human detection, leveraging transfer learning techniques to improve human

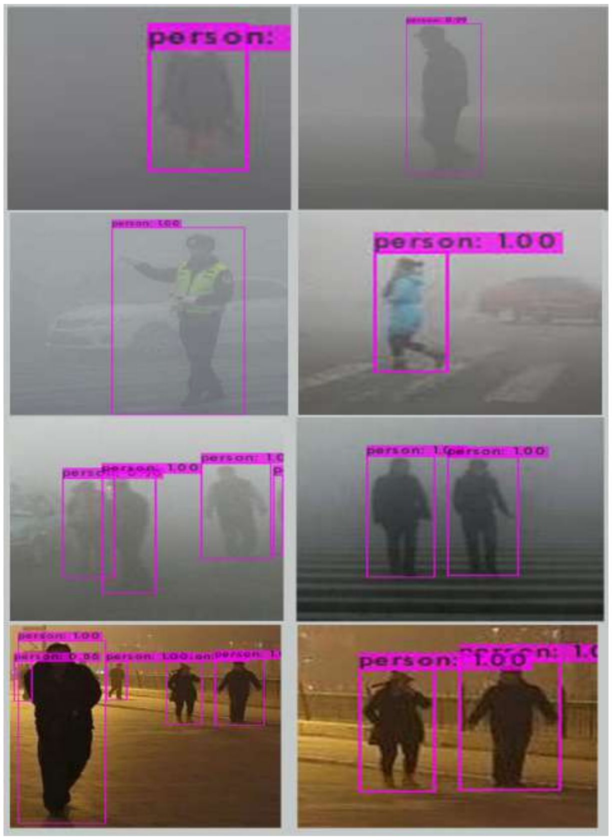


Figure 7. Detection results under challenging conditions.

detection in challenging weather conditions. Our proposed model showcased excellent results across various datasets, including INRIA, HazePerson, and Caltech. It demonstrated effective pedestrian identification even in hazy conditions while maintaining real-time processing at a speed of 55.4 FPS, highlighting its practicality and efficiency in diverse environments.

Data Availability

You can download our new annotation for the INIRA dataset by using this link. (Annotations are in YOLO annotation format): https://drive.google.com/file/d/1Ec4h1TkGtfU95kNhqF874mZMqPrLMzS7/view?usp=sharing

References

[1] R. Ayad and F. Q. Al-Khalidi, "Convolutional neural network (cnn) model to mobile remote surveillance system for home security," *International Journal of Computing and Digital Systems*, vol. 13, no. 1, pp. 1–1, 2023.

[2] M. Wafi Abdul Aziz and M. Faiz Ramli, "Region detection technique using image subtraction and pixel expansion cue for obstacle detection system on small–sized uav," *International Journal of Computing and Digital Systems*, vol. 15, no. 1, pp. 1–12, 2024.

[3] A. HATTAB and A. BEHLOUL, "A robust iris recognition approach based on transfer learning," *International Journal of Computing and Digital Systems*, 2023.

[4] S. T Padmapriya, T. Chandrakumar, and T. Kalaiselvi, "Improving the prediction accuracy of mri brain tumor detection and segmentation," *International Journal of Computing and Digital Systems*, vol. 15, no. 1, pp. 1–10, 2024.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[6] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[9] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[10] R. Joseph and A. Farhadi, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[11] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[12] G. Jocher, "Yolov5 by ultralytics," Code repository: https://github.com/ultralytics/yolov5, May 2020.

[13] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie *et al.*, "Yolov6: A single-stage object detection framework for industrial applications," *arXiv preprint arXiv:2209.02976*, 2022.

[14] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7464–7475.

[15] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics yolov8," https://github.com/ultralytics/ultralytics, 2023.

[16] R. T. Tan, "Visibility in bad weather from a single image," in *2008 IEEE conference on computer vision and pattern recognition*. IEEE, 2008, pp. 1–8.

[17] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 12, pp. 2341–2353, 2010.

[18] S.-C. Huang, J.-H. Ye, and B.-H. Chen, "An advanced single-image visibility restoration algorithm for real-world hazy scenes," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 5, pp. 2962–2972, 2014.

[19] B.-H. Chen, S.-C. Huang, and F.-C. Cheng, "A high-efficiency and high-speed gain intervention refinement filter for haze removal," *Journal of Display Technology*, vol. 12, no. 7, pp. 753–759, 2016.

[20] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human

detection using a cascade of histograms of oriented gradients," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1491–1498.

[21] H. Kieritz, S. Becker, W. Hübner, and M. Arens, "Online multi-person tracking using integral channel features," in *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2016, pp. 122–130.

[22] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.

[23] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[24] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005, pp. 886–893.

[25] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient matching of pictorial structures," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, vol. 2. IEEE, 2000, pp. 66–73.

[26] M. E. Mihçioğlu and A. Z. Alkar, "Improving pedestrian safety using combined hog and haar partial detection in mobile systems," *Traffic injury prevention*, vol. 20, no. 6, pp. 619–623, 2019.

[27] K. Kumar and R. K. Mishra, "A heuristic svm based pedestrian detection approach employing shape and texture descriptors," *Multimedia Tools and Applications*, vol. 79, pp. 21 389–21 408, 2020.

[28] Y. Xiao, K. Zhou, G. Cui, L. Jia, Z. Fang, X. Yang, and Q. Xia, "Deep learning for occluded and multi-scale pedestrian detection: A review," *IET Image Processing*, vol. 15, no. 2, pp. 286–301, 2021.

[29] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[30] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6154–6162.

[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[32] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[33] J. Wu, C. Zhou, Q. Zhang, M. Yang, and J. Yuan, "Self-mimic learning for small-scale pedestrian detection," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2012–2020.

[34] C. Kyrkou, "Yolopeds: efficient real-time single-shot pedestrian detection for smart camera applications," *IET Computer Vision*, vol. 14, no. 7, pp. 417–425, 2020.

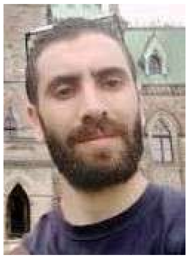
[35] M.-L. Li, G.-B. Sun, and J.-X. Yu, "A pedestrian detection network model based on improved yolov5," *Entropy*, vol. 25, no. 2, p. 381, 2023.

[36] L. Huang, Z. Wang, and X. Fu, "Pedestrian detection using retinanet with multi-branch structure and double pooling attention mechanism," *Multimedia Tools and Applications*, vol. 83, no. 2, pp. 6051–6075, 2024.

[37] E. Bisong and E. Bisong, "Google colaboratory," *Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners*, pp. 59–64, 2019.

[38] AlexeyAB, "darknet - how to train to detect your custom objects," 2020. [Online]. Available: https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects

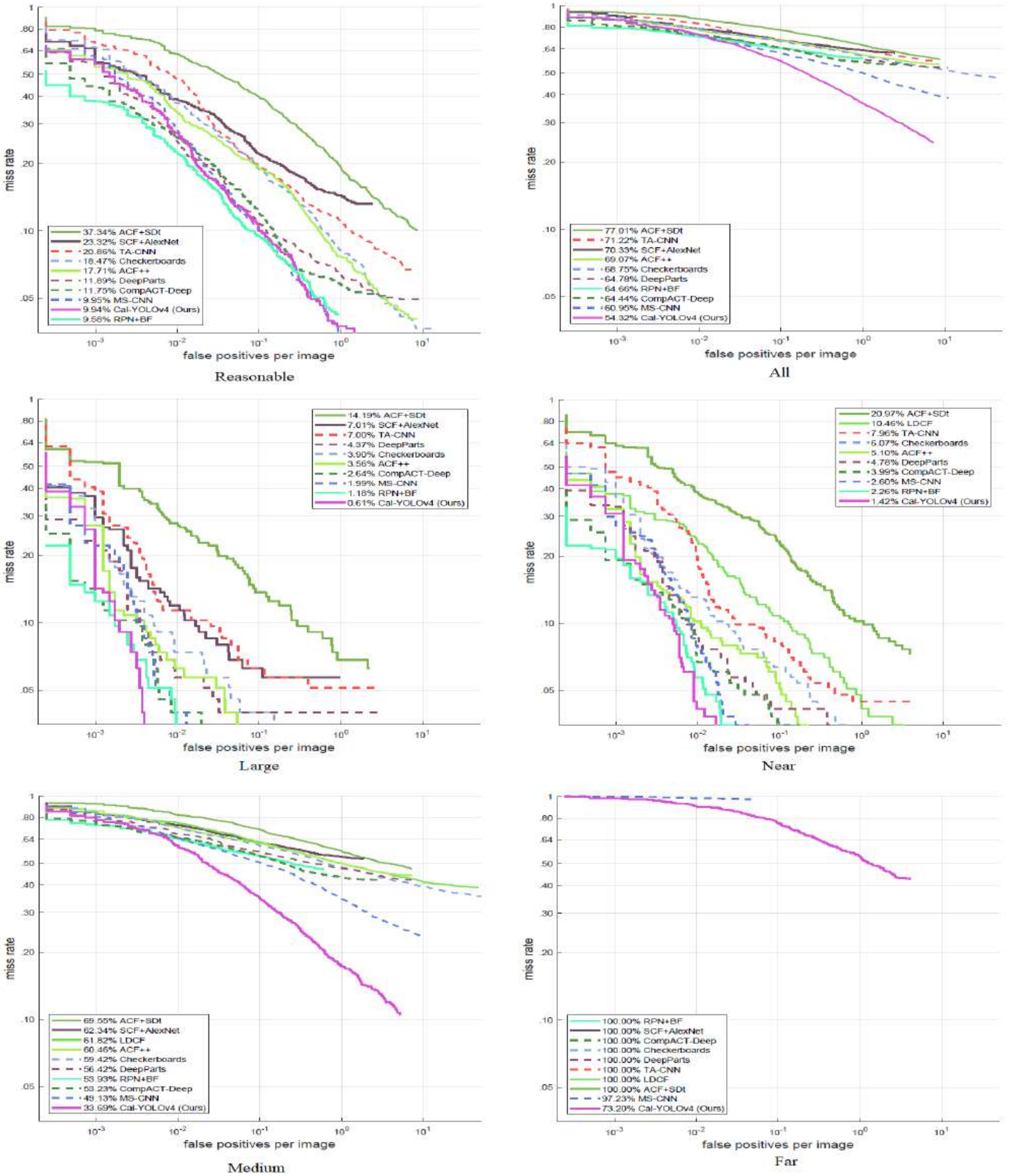
[39] G. Li, Y. Yang, and X. Qu, "Deep learning approaches on pedestrian detection in hazy weather," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8889–8899, 2019.



Yassine Bouafia Is a PhD student in the computer science department at University of Batna 2, Algeria. He obtained a Master's degree in computer science in 2016 from the University of Batna 2. He is a member of the LaSTIC Laboratory. His research interests are in artificial intelligence, computer vision, object detection, image classification, and deep learning.

Larbi Guezouli received the Ph.D. degree in computer science from the University of Paris 7 (Denis Diderot), France. He is a teacher-researcher at the Higher National School of Renewable Energy, Environment and Sustainable Development, Batna, Algeria. His research interests are in information retrieval, data mining, cross-language, machine learning, artificial intelligence, Big data, and IoT.

Hicham Lakhlef Is an associate professor at the University of Technology of Compiegne (UMR CNRS 7253). He obtained his Ph.D. degree from the University of FrancheComté in 2014. He obtained his Master's degree from the University of Picardie Jules Verne in 2011. His research interests are in parallel and distributed algorithms, WSNs, clustering, optimization, routing, and IoT.



Reasonable

All

Large

Near

Medium

Far

Figure 8. Evaluation results under six different evaluation settings on the Caltech Pedestrian Data Set. Reasonable, All, Large, Near, Medium, Far