MJFAS
Malaysian Journal of Fundamental and Applied Sciences

RESEARCH ARTICLE

Effect of Positive-Negative Image Ratio on the Performance of Pedestrian Detection Model

Lai Kok Yee[a], Tan Lit Ken[a]*, Hau Sim Choo[a], Yutaka Asako[a], Lee Kee Quen[a], Hooi-Siang Kang[b], Y. S. Gan[c], Zun-Liang Chuan[d], Wah Yen Tey[e], Nor Azwadi Che Sidik[a]

[a]Malaysia–Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, 54100 Kuala Lumpur, Malaysia; [b]Marine Technology Center, Institute for Vehicle System & Engineering, School of Mechanical Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia; [c]School of Architecture, Feng Chia University, Taichung 40724, Taiwan R.O.C; [d]Faculty of Industrial Sciences and Technology, Universiti Malaysia Pahang; [e]Department of Mechanical Engineering, UCSI University, Cheras, Kuala Lumpur, Malaysia

Abstract Pedestrian detection holds significant importance in computer vision, finding applications in video surveillance, human-computer interaction, and autonomous vehicles. Surprisingly, there is a scarcity of research addressing the optimal ratio of positive to negative images for training detection models. This study endeavors to fill this research gap by exploring various detection models and determining the ideal ratio. Two distinct scenarios are investigated, each characterized by an equal total image count and an equivalent number of positive images sourced from CVC-14 night/visible, night/FIR, and INRIA databases. The study leverages the Histogram of Oriented Gradient, utilizing both Support Vector Machines and Medium Neural Networks to construct the detection models. Notably, the experiments reveal that the accuracy of the models remains relatively stable, even with an increase in the ratio of negative images. However, a noteworthy inverse relationship between sensitivity and specificity emerges as the ratio escalates. The findings, guided by the Youden Index, pinpoint the optimal training ratio for pedestrian detection models, falling within the range of 1:0.5 to 1:2. In the CVC-14 nighttime database, the Youden index reached 100% when the model was trained with a 1:0.5 ratio using SVM, and a total of 4500 images were employed in the training process. On the other hand, in the INRIA dataset, the Youden index exhibited its highest value at 98.50%. This occurred when both SVM and a Medium neural network were utilized to train the model with a ratio of 1:2, utilizing a total of 3000 images for the training phase. It's worth highlighting that the processing time for SVM models lags behind that of Medium Neural Networks. This disparity arises from the heightened computational complexity inherent to medium-sized neural networks, making them computationally demanding compared to SVMs. This study contributes valuable insights into the nuanced relationship between image ratios and the performance of pedestrian detection models.

Keywords: Pedestrian detection, ratio, Histogram of Oriented Gradient, Support Vector Machines, Medium Neural Network.

***For correspondence:**
tlken@utm.my

Received: 23 Nov. 2023
Accepted: 4 Feb. 2024

Introduction

In recent years, motion detection has been a hot topic in scientific research and engineering applications [1-4]. In motion detection, detecting humans accurately is crucial in visual surveillance and person

10.11113/mjfas.v20n2.3300

identification. However, vision-based detection is still very challenging due to the diverse kinds of apparel, lighting environments, complicated environments, occlusion effects, and a broad range of human poses and views [5-8]. Infrared images are superior to visible images as they can penetrate obscurants such as fog, mist, aerosols, etc., more effectively in poor atmospheres.

Features and classifiers are the centers of attention in pedestrian detection research. Various feature descriptors and classifiers have been developed to encounter different situations. Histogram of Oriented Gradient (HOG) is created as a shape descriptor [9] and Local Binary Pattern (LBP) is created to represent the texture of the images [10]. Furthermore, these descriptors are enhanced to face various situations such as occlusion, illumination, etc. For classifiers, various algorithms are created to classify the features based on different focuses such as human body parts, clustering, hyperplane, and more.

To effectively train a pedestrian detection algorithm, the availability of positive images (depicting scenes with pedestrians) and negative images (without pedestrians) is crucial. In conventional research practices, negative images are often synthesized randomly from the background of existing images [11]. While this approach simplifies the acquisition of negative samples, it has inadvertently led to an escalation in the ratio of negative images in recent studies. As this ratio increases, the detection model's accuracy may improve, but it also introduces the risk of biasing the model towards the negative class due to an imbalance in the dataset [12]. The augmentation of negative images can result in a model that is overly inclined to predict the absence of pedestrians, essentially favoring the majority class. This bias occurs as the model learns to minimize overall error by predominantly classifying instances as the prevalent class. In the context of pedestrian detection, such bias could lead to a model consistently predicting "no pedestrian," resulting in missed detections. Recognizing this challenge, it becomes imperative to adjust the ratio of positive to negative samples during the training process. This adjustment serves as a countermeasure to mitigate biases and prevent the model from being overly influenced by the abundance of negative instances. Striking an optimal balance in the ratio ensures that the model learns to discern both positive and negative cases effectively, enhancing its ability to make accurate predictions across a spectrum of real-world scenarios. Consequently, this nuanced approach contributes to the creation of a more robust and unbiased pedestrian detection model. In this paper, the primary contribution of this research lies in investigating the impact of the ratio of positive to negative images on training a human detection model. Given the absence of prior studies addressing the influence of this ratio on human detection models, we believe that identifying the optimal ratio is of paramount significance, as it directly affects the models' performance.

The paper is organized as follows. Section 2 discusses the related work regarding pedestrian detection. Section 3 introduces the method proposed to identify the optimum ratio between positive and negative images for the training detection model. Section 4 shows the experimental results and discussion of the results. Finally, section 5 summarizes this paper.

Literature Review

Conventional methods have always been a hot research topic in pedestrian detection. The conventional method of pedestrian detection consists of two steps: feature extraction and classification. Feature descriptors such as Histogram of Oriented Gradient (HOG) [13], Local Binary Pattern (LBP) [14], Scale Invariant Features Transform (SIFT) [15], Haar-like features [16], DPM (Deformable Parts Model), SURF (Speeded Up Robust Features), and the Viola-Jones algorithm have played significant roles in shaping the landscape of pedestrian detection.

Histogram of Oriented Gradient (HOG) features serve as shape-based representations, capturing the orientation histogram of edge intensities. These features construct histograms within sub-blocks of an image based on the strength of gradient information and accumulation of direction at each pixel within the sub-block. HOG excels in portraying a pedestrian's shape and appearance with high accuracy.

In contrast, appearance features like Local Binary Pattern (LBP) and Haar-like features focus on extracting texture and color information from local images. LBP features, while computationally efficient with high discriminative power, are less suitable for pedestrian detection in complex backgrounds due to their threshold function [17].

The Deformable Parts Model (DPM) [18] offers an alternative approach by decomposing objects into parts, providing flexibility in capturing variations in object pose and appearance. However, DPM is computationally expensive and may struggle with extremely small or highly occluded objects.

The Viola-Jones algorithm [19], an influential early method, relies on Haar-like features for rapid object detection. Characterized by its use of integral images and an AdaBoost-based classifier, Viola-Jones has been historically significant, particularly in face detection applications. However, its limitation to rectangular Haar-like features may impede its ability to capture complex patterns well. Additionally, it is sensitive to variations in scale and rotation.

Speeded Up Robust Features (SURF) [20] is another feature descriptor known for its speed and robustness to transformations. Capable of efficiently extracting distinctive features, SURF has found use in pedestrian detection. However, it is memory-intensive and may not perform well with repeated patterns.

HOG proposed by Dalal and Triggs has been proven to be the most effective feature for pedestrian detection among all the other feature descriptors [13]. However, HOG features-based detection results in poor real-time performance due to its high computational complexity. Many methods based on HOG have been invented to improve the performance of HOG. Qiang *et al*. integrated the cascade-of-rejectors concept with HOG to achieve fast and true human detection [21]. A large set of blocks at multiple sizes, locations, and aspect ratios is used for feature selection in this method. Ning He *et al*. designed a new feature descriptor called Scale Space Histogram of Oriented Gradients (SS-HOG) [22]. By integrating scale space theory with HOG, this method encodes the information of body contour at multiple scales compared to the original HOG. Tomoki Watanabe *et al*. proposed Co-occurrence histograms of oriented gradients (CoHOG) by using pairs of gradient orientations as units to build histograms [23]. This method can express local and global shapes in detail but ends up with a high number of feature dimensionality. Masayuki *et al*. improved CoHOG by dividing it into small features and combining many weak classifiers to create a cascade classifier [24]. These variants of HOG show the potential of HOG feature as a pedestrian descriptor. Therefore, HOG is selected as the pedestrian descriptor in this research.

For classification, R. Quintero *et al*. developed Hidden Markov Model (HMM) that recognizes pedestrian intention using 3D position and displacements of 11 joints located along the bodies [25]. However, the performance of this method is affected by human activities such as self-occlusion. Therefore, Kamal *et al*. improved this method by modifying the hidden Markov Model (M-HMM) to classify human activities based on human body parts rather than using body joints [26]. Youv and Robert introduced Adaboost algorithm to classify humans into a group compared to other objects [27]. This algorithm will learn from labeled data with observed output to make predictions. A weak classifier method is chosen in each round to reduce the optimal training error. Therefore, this algorithm is a popular boosting algorithm that can optimize the classification error. For classification and recognition, Support vector machine (SVM) is trained with positive and negative images and then gives a test sample for testing. SVM constructs a set of hyperplanes in an infinite dimensional space to classify pedestrians [28]. It tracks the problem by representing the data in the higher dimensional space and makes the classification easier in space. SVM is the most utilized method for human classification and activity recognition.

In contrast to the traditional HOG-SVM approach, Convolutional Neural Networks (CNNs) have demonstrated their superiority in efficiently extracting high-level contour features and achieving state-of-the-art performance in real-time multiple-object tracking (MOT) [29]. CNN-based models primarily focus on local feature extraction, often overlooking global features [30]. To address this limitation, a hybrid approach combining HOG and CNN has been proposed to enhance the performance of detection models.

In 2016, Zhang *et al*. utilized HOG to eliminate background noise when constructing detection models for moving objects in videos, resulting in excellent performance for detecting moving objects in applications such as food and agricultural traceability analysis [31]. Building upon this foundation, Lipetski *et al*. introduced the HCNN model by integrating the HOG descriptor with CNN to enhance the quality of pedestrian detection [30]. Rui *et al*. presented an algorithm that leverages various feature maps from the initial CNN layer as input to HOG, demonstrating that combining HOG-based multi-convolutional features for pedestrian detection can yield high accuracy and stable network performance [32].

Different ratio of positive images and negative images is used for different researchers. This caused the results for each researcher to vary from each other. For example, in Dalal *et al*. research, the ratio of 1:5 between positive and negative images is used to train the HOG-SVM model [13]. 2478 positive images and 12180 negative are used and the result shows 89% at $10^{-4} FPPW$ (false positive per window). Ni Chen and his team proposed a HOG-SVM with differential evolution model using the ratio of 1:1 where 150 images for each positive and negative image from the INRIA dataset [33]. The detection rate of the model created by Ni Chen *et al*. is 80%. For Haythem *et al*., they used the ratio of 1:0.5 which is 2436 positive images and 1218 negative images to train a HOG model [34]. The detection rate of the model is 85.9%. These showed that the ratio of positive and negative images used for training detection model has a great effect on the detection rate. However, no research has been done on the ratio between positive and negative images used to train the pedestrian detection model. Therefore, the optimum ratio of positive and negative images for model training is unknown.

In this study, the Histogram of Oriented Gradient (HOG) is specifically chosen as the feature descriptor for pedestrian detection. HOG exhibits a remarkable ability to encode characteristic patterns of human shapes and textures, imparting robustness to variations in pose, scale, and illumination. Its effectiveness in capturing relevant information from images makes it well-suited for discerning pedestrians amidst diverse environmental conditions. Simultaneously, for the classification component of the model, Support Vector Machine (SVM) and Convolutional Neural Network (CNN) is adopted. SVM, a powerful binary classifier, is selected for its capability to establish a global decision boundary that effectively separates human entities from background clusters. This aids in providing a robust and accurate overall classification. CNNs are integrated into the model due to their proven superiority in extracting high-level features from images. The study further explores the impact of various ratios of positive to negative images during the model training phase. Multiple ratios are systematically tested, and a detailed comparison is conducted to pinpoint the optimum ratio for crafting an effective pedestrian detection model. This empirical approach ensures that the model's training is fine-tuned to strike the ideal balance, maximizing its accuracy and robustness across diverse scenarios.

Materials and Methods

Figure 1 shows the flow chart of the methodology in this research, which consists of data preparation, feature extraction, model training, model testing and performance evaluation.

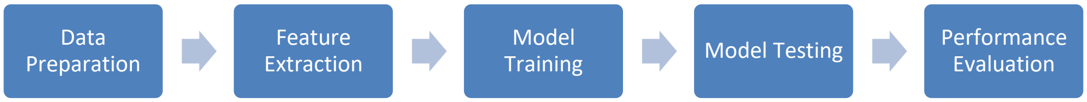


Figure 1. Flow chart of the research

Data Preparation

Two distinct datasets were employed in this study: the INRIA database and the CVC-14 database. The INRIA database, crafted by Navneet Dalal and his team [13], encompasses 1805 images of humans. Each image is standardized to dimensions of 64 * 128 pixels and originates from a diverse assortment of personal photographs. Among these images, a subset of 1200 was intentionally selected and mirrored, effectively doubling the dataset and yielding a total of 2400 positive images.

The CVC-14 database created by Alejandro Gonzalez and his team [35] is used for training and testing the detection model. The database contains four sequences of images which are day/FIR, night/FIR, day/visible, and night/visible images. In this research, night/FIR and visible images are selected to build the detection model. A FLIR Tau 2 Camera with the specification of 640 * 512 pixels was used to capture the FIR images. At the same time, IDS UI-3240CP is used to capture visible images. The acquisition of images from both modalities was performed at a frame rate of 10 FPS.

This research examined two distinct scenarios, each involving the generation of six sets of data with varying ratios of positive and negative images: 1:0.25, 1:0.5, 1:1, 1:2, 1:5, and 1:10. In the first case, we divided a total of 3000 images from INRIA and 4500 FIR images along with 3500 visible images from CVC-14 according to the specified ratio. These sets were then employed to develop the model. In contrast, the second case utilized 2400 positive images from INRIA and 3600 FIR positive images, and 2800 visible positive images from CVC-14 for each model. The number of negative images varied according to the ratios.

To create the negative image sets used in model training, we extracted images from sequences that did not contain any positive samples. For the sake of feature extraction in the subsequent steps, all images were standardized to a resolution of 64 * 128 pixels. Figure 2, Figure 3 and Figure 4 display sample images used in this research, while Table 1 provides detailed information regarding the composition of images for each case.

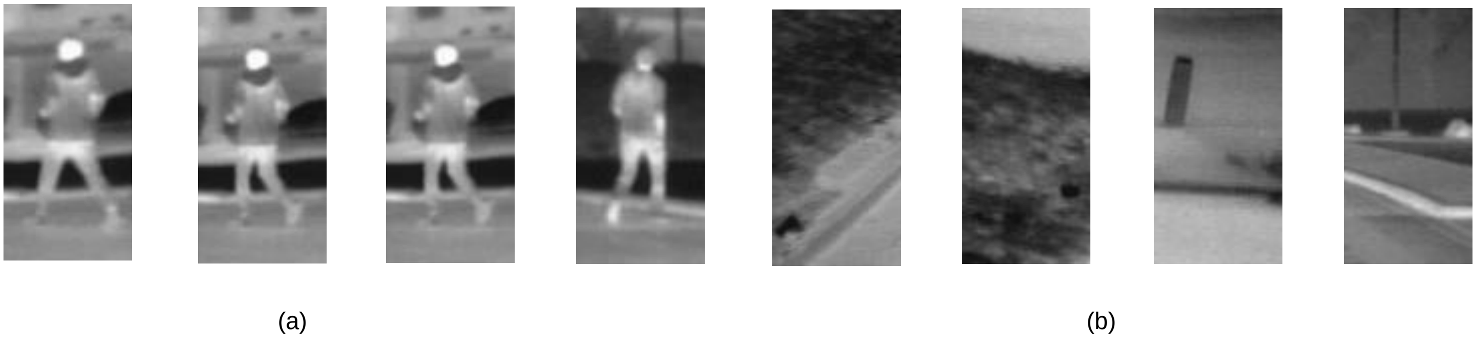

(a) (b)

Figure 2. Example of (a) positive images and (b) negative images for CVC-14 night/FIR images

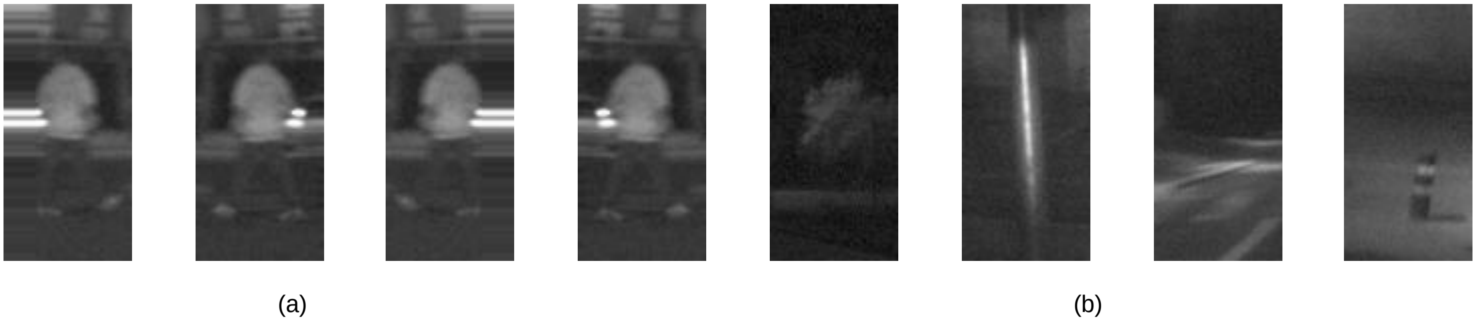

(a) (b)

Figure 3. Example of (a) positive images and (b) negative images for CVC-14 night/visible images

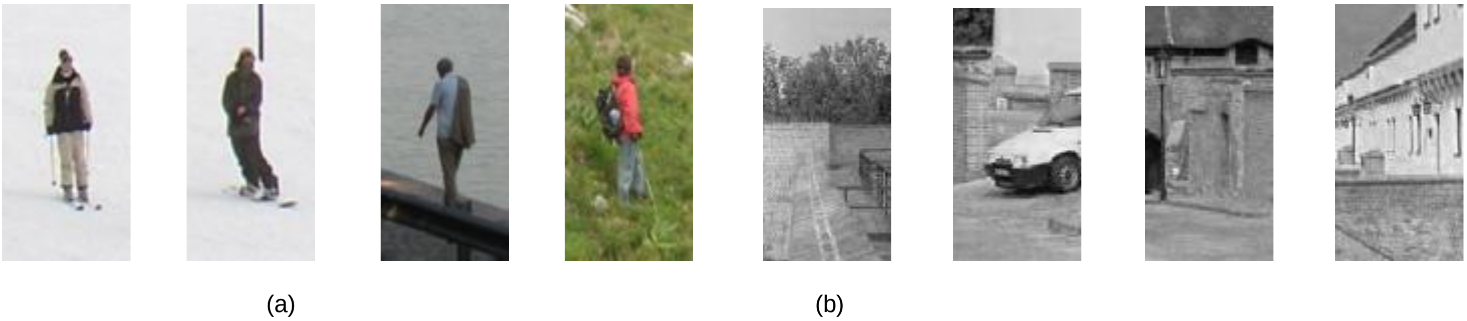

(a) (b)

Figure 4. Example of (a) positive images and (b) negative images for images from INRIA

Table 1. Total positive and negative images in (a) Case 1 and (b) Case 2 with the ratio of 1:0.25, 1:0.5, 1:1, 1:2, 1:5, and 1:10

	Ratio	**1:0.25**	**1:0.5**	**1:1**	**1:2**	**1:5**	**1:10**
CVC-14 FIR	Positive image	3600					
	Negative image	900	1800	3600	7200	18000	36000
CVC-14 Visible	Positive image	2800					
	Negative image	700	1400	2800	5600	14000	28000
INRIA	Positive image	2400					
	Negative image	600	1200	2400	4800	12000	24000

(a)

	Ratio	**1:0.25**	**1:0.5**	**1:1**	**1:2**	**1:5**	**1:10**
CVC-14 FIR	Positive image	3600	3000	2250	1500	750	409
	Negative image	900	1500	2250	3000	3750	4091
CVC-14 Visible	Positive image	2800	2335	1750	1165	585	320
	Negative image	700	1165	1750	2335	2915	3180
INRIA	Positive image	2400	2000	1750	1000	500	272
	Negative image	600	1000	1750	2000	2500	2728

(b)

Feature Extraction

HOG descriptor is used to extract information about the contour of objects present in the images. There are 5 steps in extracting HOG feature from images which are listed below:

1. Image preprocessing: the only constraint for the HOG descriptor is that the images being analyzed have a fixed aspect ratio which is 1:2. According to Dalal *et al*. research, the HOG descriptor provides the best results when the images used for feature extraction are in size of 64*128 resolution. Therefore, in this research, all the images used are resized into this aspect of resolution before extracting the feature.

2. Gradient computation: to achieve the HOG, the horizontal and vertical gradients are calculated by filtering the image with the following kernels: [-1,1], while the magnitude and direction of the gradient is calculated using the following formula.

$$(x_i\ ,\ y_i\ ) = \sqrt{\partial_x(x_i\ , y_i)^2 + \partial_y(x_i\ ,\ y_i)^2}$$

$$\theta(x_i\ , y_i\ ) = \arctan\left(\frac{\partial_y\ (x_i\ , y_i\ )}{\partial_x\ (x_i\ , y_i\ )}\right)$$

Where m is the magnitude of the gradient and theta is the direction of the gradient.

3. Weighted vote into cells: The image is divided into 8*8 cells and histogram of gradient is obtained from each cell. Then, a histogram containing 9 bins corresponding to angles 0, 20, 40…160 is created for each cell. The histogram of gradients obtained from each cell is then rearranged into the 9 bins histogram based on its direction and magnitude.

4. Normalization: In this step, the histogram is normalized so it will not be affected by lighting variations. L2 norm is used to identify the length of the vector and each value of the histogram is normalized by dividing the value of the length of the vector. The equation below shows the formula of normalization:

$$v \rightarrow v/\sqrt{||v||_2^2 + \varepsilon^2}$$

Where v represents histogram of vector of a block before normalization and ε is a minimal constant to prevent zero division error.

5. HOG descriptor: At this final step, all the histogram is concatenated to form the final feature vector. For each image, a histogram with a 3780-dimensional vector will be formed. Figure 5 illustrates the concatenation of the histogram.

.

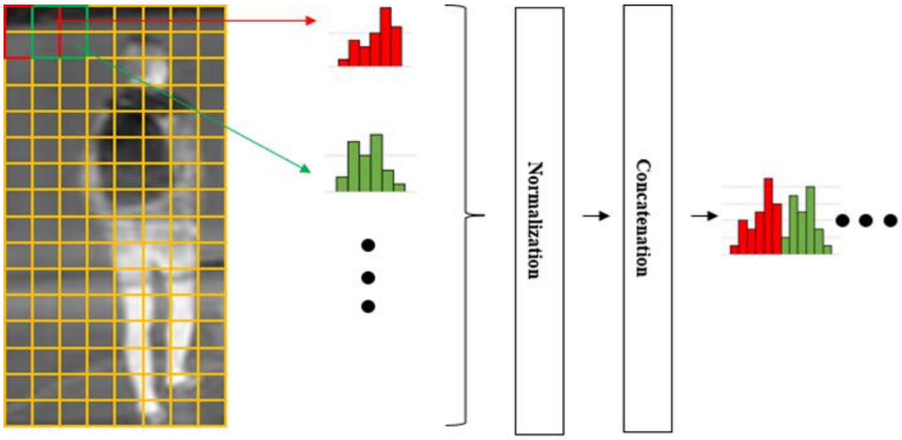


Figure 5. Illustration of the concatenation of histogram inside HOG

Data Training

In this section, we partition the entire dataset using an 80:20 ratio. Specifically, 80% of the images from the database are allocated for training purposes, leaving the remaining 20% for performance evaluation. To construct the detection models, we employ the histogram generated in the preceding section as input for both SVM and neural network classifiers.

For SVM, we opt for a Linear SVM due to its simplicity and ease of comprehension compared to more intricate neural networks. Linear SVMs yield results that are interpretable and amenable to visualization in lower-dimensional spaces. Additionally, they demonstrate a reduced susceptibility to overfitting, rendering them suitable for scenarios where data is limited.

On the other hand, we choose a medium-sized neural network for our neural network classifier. These networks strike a balance between overly simplistic models that may underfit the data and excessively complex ones that demand substantial computational resources.

In conducting the training and testing processes with MATLAB version 2023a, we leverage the default hyperparameters offered by the MATLAB platform for the classifiers. The entire computational workflow unfolds on a personal computer equipped with robust hardware, featuring an AMD RYZEN 5 3600 processor, 16GB RAM, and a NVIDIA GALAX RTX2060 GPU. This configuration ensures a powerful and efficient environment for the execution of machine learning tasks, facilitating the training of models and subsequent evaluations within a seamlessly integrated MATLAB environment. Table 2 furnishes a summary of the hyperparameters employed during the training of the detection models for each classifier.

Table 2. Hyperparameters for (a) Linear SVM and (b) Medium Neural Network

Classifier	**SVM**
Kernel function	Linear
Kernel equation	$k(x, y) = x \cdot y$
Box constraint level	1
Multiclass method	One-vs-one

(a)

Classifier	**Medium Neural Network**
Number of fully connected layers	1
First layer size	25
Activation	ReLU
Iteration limit	1000
Lambda	0

(b)

Performance Evaluation

The performance of the models was evaluated by accuracy, sensitivity, and specificity. The sensitivity of the model is the proportion of total humans detected among those images with humans [36]. Meanwhile, the specificity of the model is the proportion of the detected non-human images among the images without humans [36]. Sensitivity and specificity are chosen as evaluation metrics for the human detection model, primarily due to the elevated costs associated with both false positives and false negatives. In the context of human detection, the repercussions of failing to identify a human (false negative) and triggering a false alarm (false positive) can be substantial. By prioritizing sensitivity, the model aims to minimize the instances of undetected humans, ensuring a reduced risk of overlooking potential threats. Simultaneously, an emphasis on specificity helps mitigate false alarms, which could lead to unnecessary actions. The formula for accuracy, sensitivity, and specificity are shown below:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{FP + TN}$$

Where TP is defined as true positive, TN is defined as negative, FP is defined as false positive and FN is defined as false negative.

Striking a balance between sensitivity and specificity is crucial to optimize the model's performance, aligning with the objective of achieving accurate and reliable human detection while minimizing the associated costs of errors. The Youden Index, denoted by J [37], serves as a unified metric that integrates both sensitivity and specificity into a single measure. It is employed to determine the optimal cut-off point in a classification model. By encapsulating both sensitivity and specificity, the Youden Index provides a comprehensive evaluation, enabling the identification of the most effective threshold that strikes a balance between minimizing false positives and false negatives. Youden index is defined as follows:

$$J = Max_c(Sensitivity_c + Specificity_c - 1)$$

Where c is defined as the optimal cutoff.

Results and Discussion

CVC-14 Night/FIR images

Comparison of the Same Number of Images during Training Stage

The experiment involved six distinct image datasets, each featuring varying ratios, to conduct a comprehensive analysis. As illustrated in Table 3, both classifiers consistently demonstrated remarkable accuracy levels across all data ratios. It is noteworthy that the linear SVM, in some instances, exhibited a slightly lower accuracy compared to the medium neural networks. Specifically, SVM models achieved accuracy ranging from 99.20% to a perfect 100%, while medium neural network models demonstrated accuracy between 99.20% and 99.80%. This underscores the robust and consistent accuracy performance of both classifiers across the spectrum of data ratios.

Within the SVM models, sensitivity exhibited an intriguing pattern. It surged from 99.72% at the 1:0.25 ratio to an impeccable 100% at the 1:0.5 ratio, only to gradually decline to 93.83% as the ratio increased to 1:10. In contrast, specificity commenced at 98.33% for the 1:0.25 ratios and steadily ascended to reach 100% at the 1:5 ratio.

For the medium neural network, sensitivity displayed a decreasing trend, diminishing from 99.72% at the 1:0.25 ratio to 96.30% at the 1:10 ratio. Conversely, specificity started at 98.89% for the 1:0.25 ratio, peaked at 100% for the 1:2 ratio, and subsequently receded to 99.87% for the 1:5 ratio and 99.76% for the 1:10 ratio. Notably, an inverse relationship between sensitivity and specificity within both SVM and medium neural network models was observed.

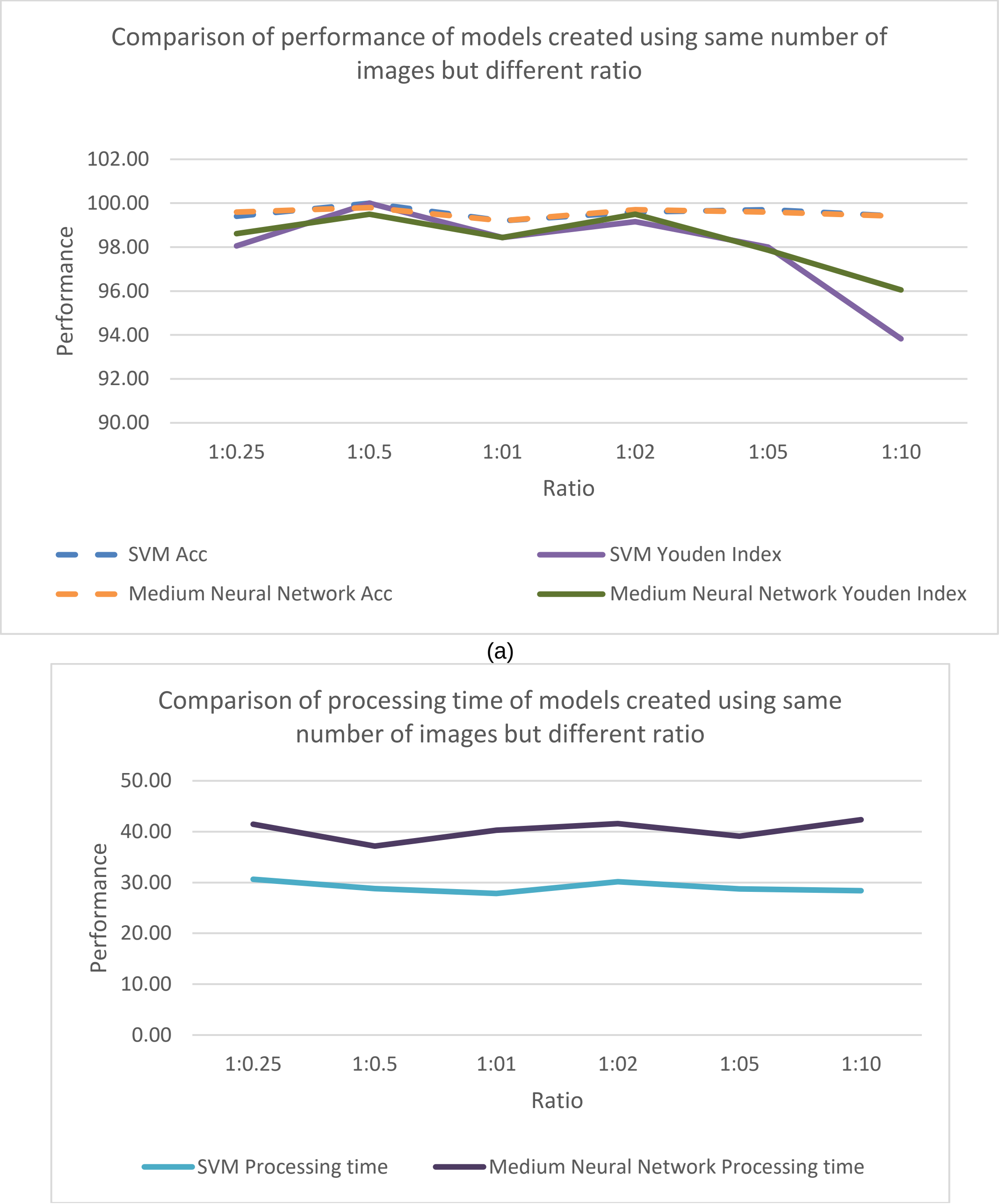
To assess discrimination performance comprehensively, sensitivity and specificity were amalgamated into a single metric, the Youden Index (J). In Figure 6(a), the SVM model achieved its highest J value at

100% for the 1:0.5 ratio, while the medium neural network attained the same 99.5% J value for the 1:0.5 and 1:2 ratios. Notably, the J values for both SVM and medium neural network models exhibited fluctuations at the 1:1 ratio, reaching 98.44% for both. This phenomenon can be attributed to models tending to overfit when classes are perfectly balanced during training, posing challenges for machine learning models not explicitly designed to handle class imbalance. Overfitting occurs when models learn to fit noise rather than capture underlying patterns, potentially leading to diminished generalization performance on unseen data.

Additionally, it's worth noting that Linear SVM consistently demonstrated shorter processing times in comparison to the Medium Neural Network. Processing time for SVM models ranged from 27.84 seconds to 30.63 seconds, while for Medium Neural Network, it spanned from 37.16 seconds to 42.35 seconds, as depicted in Figure 6(b). This efficiency renders Linear SVM particularly appealing for scenarios where computational resources or stringent time constraints are pivotal factors in model selection.

Table 3. Result of the models trained with different positive-negative ratios using same amount of Night/FIR images

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
SVM	Acc	99.40	100.00	99.20	99.60	99.70	99.40
	Sensitivity	99.72	100.00	99.56	99.50	98.00	93.83
	Specificity	98.33	100.00	98.89	99.67	100.00	100.00
	Youden Index	98.06	100.00	98.44	99.17	98.00	93.83
	Processing time (s)	30.63	28.79	27.84	30.15	28.72	28.38
Medium Neural Network	Acc	99.60	99.80	99.20	99.70	99.60	99.40
	Sensitivity	99.72	99.67	99.11	99.50	98.00	96.30
	Specificity	98.89	99.83	99.33	100.00	99.87	99.76
	Youden Index	98.61	99.50	98.44	99.50	97.87	96.05
	Processing time (s)	41.46	37.16	40.30	41.59	39.13	42.35



(a)

(b)

Figure 6. Comparison of (a) performance and (b) processing time of models trained using same number of Night/FIR images

Comparison of the Same Number of Positive Images during Training Stages

The fundamental algorithms employed in SVM and neural networks diverge significantly. SVMs employ convex optimization techniques known for their efficiency in seeking the global minimum. Conversely, neural networks rely on gradient-based optimization methods, which entail a greater number of iterations and computational operations. These algorithmic distinctions contribute to the variance in processing time between the two models.

CVC-14 Night/Visible Images

Comparison of the Same Number of Images during Training Stage

In Table 4, both classifiers consistently demonstrated outstanding levels of accuracy across a range of data ratios. It's worth noting that the linear SVM, on occasion, exhibited a slightly lower accuracy when compared to the medium neural networks. To provide precise figures, SVM models achieved accuracy rates spanning from an impressive 99.40% to an exceptional 99.90%, while medium neural network models showcased accuracy levels ranging between 99.50% and 99.90%.

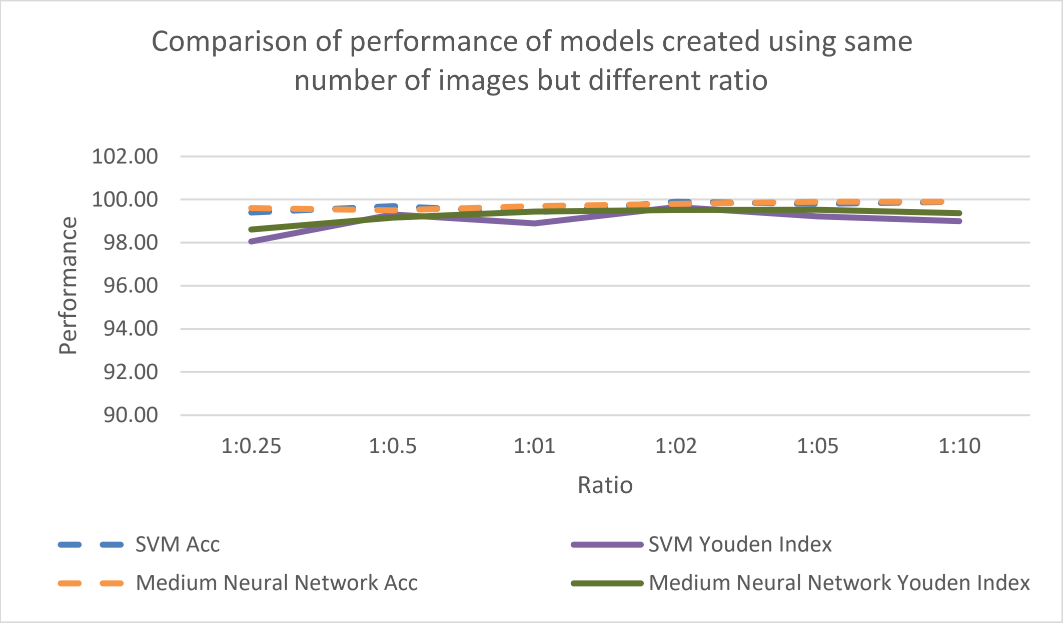
Within the SVM models, sensitivity displayed a distinctive pattern. It started at 99.72% for the 1:0.25 ratio, climbed to 99.86% at the 1:0.5 ratio, and then experienced a dip to 99.58% before rebounding to 99.72% at the 1:2 ratio. However, sensitivity gradually declined as the data ratio increased, reaching 99.31% at the 1:5 ratio and ultimately settling at 99.03% for the 1:10 ratio. Meanwhile, the specificity of the SVM model consistently improved, ascending from 98.33% (1:0.25 ratio) to 99.44% (1:0.5 ratio), 99.93% (1:2 ratio), 99.92% (1:5 ratio), and finally reaching an outstanding 99.97% at the 1:10 ratio. Notably, the model's specificity experienced a temporary decline to 99.31% at the 1:1 ratio, attributed to overfitting.

In the case of the medium neural network, sensitivity exhibited a distinct trend, declining from 99.72% (1:0.25 ratio) to 99.44% for both the 1:0.5 ratio and the 1:1 ratio. Subsequently, it saw a slight rise to 99.58% at the 1:2 ratio and the 1:5 ratio, only to decrease again to 99.44% for the 1:10 ratio. Meanwhile, specificity increased from 98.89% to 99.72% and ultimately reached a perfect 100% for the 1:1 ratio. As the data ratio increased, specificity gradually decreased to 99.93% (1:2 ratio and 1:10 ratio) and 99.94% (1:5 ratio).

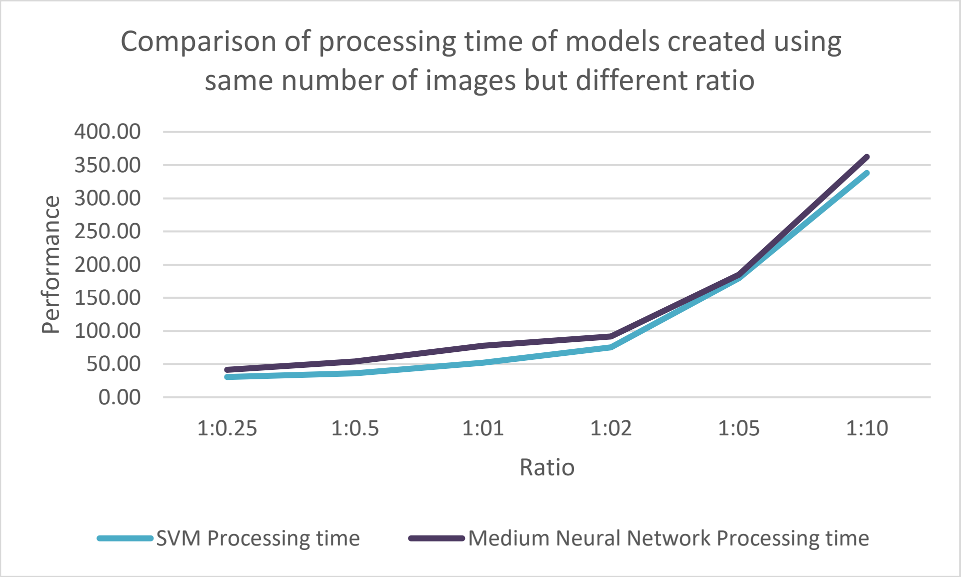
As depicted in Figure 7(a), the SVM models exhibit their peak J value, achieving a remarkable 99.65% at the 1:2 ratio. Conversely, for the medium neural network models, the J value reaches its zenith when the model's ratio is set at 1:5, attaining an impressive 99.53%. In general, the J values for medium neural network models surpass those of SVM models, except for models constructed at the 1:0.5 and 1:2 ratios. As illustrated in Figure 7(b), the processing time for the medium neural network is marginally higher than that of the SVM model. Notably, both models exhibit an increase in processing time as the training ratio increases. The primary explanation for the medium neural network models requiring more processing time lies in the neural network's heightened complexity, demanding training prerequisites, and the inherent disparities in their algorithms and implementations.

Table 4. Result of the models trained with different positive-negative ratios using same amount of positive Night/FIR images.

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
SVM	Acc	99.40	99.70	99.40	99.90	99.80	99.90
	Sensitivity	99.72	99.86	99.58	99.72	99.31	99.03
	Specificity	98.33	99.44	99.31	99.93	99.92	99.97
	Youden Index	98.06	99.31	98.89	99.65	99.22	99.00
	Processing time (s)	30.63	36.39	51.88	75.47	179.52	338.18
Medium Neural Network	Acc	99.60	99.50	99.70	99.80	99.90	99.90
	Sensitivity	99.72	99.44	99.44	99.58	99.58	99.44
	Specificity	98.89	99.72	100.00	99.93	99.94	99.93
	Youden Index	98.61	99.17	99.44	99.51	99.53	99.38
	Processing time (s)	41.46	54.28	77.76	91.52	184.68	362.45



(a)



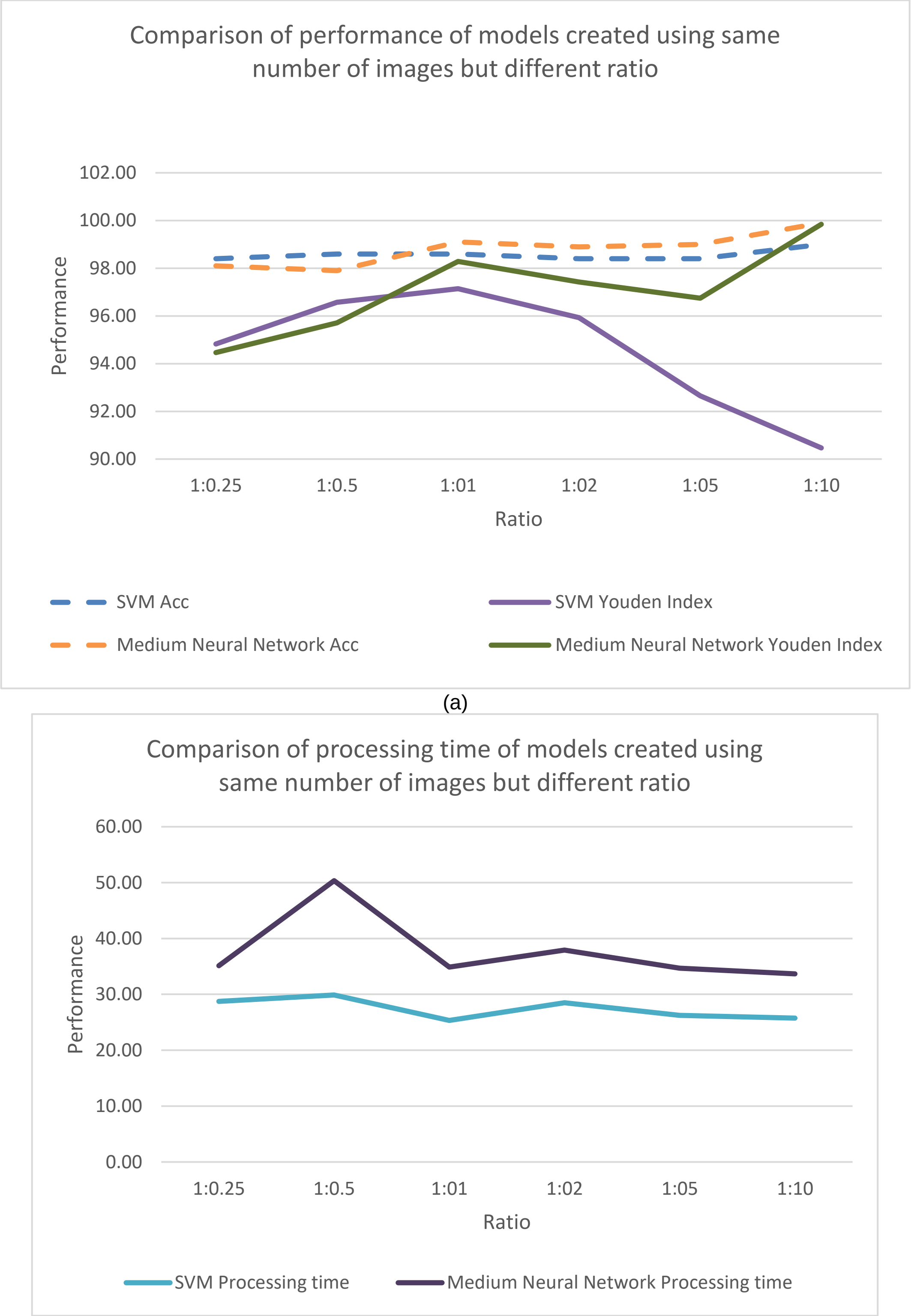
(b)

Figure 7. Comparison of (a) performance and (b) processing time of models trained using a same number of positive Night/FIR images

Table 5. Result of the models trained with different positive-negative ratios using a same number of Night/Visible images

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
SVM	Acc	98.40	98.60	98.60	98.40	98.40	99.00
	Sensitivity	99.11	99.14	99.43	96.57	93.16	90.63
	Specificity	95.71	97.42	97.71	99.36	99.49	99.84
	Youden Index	94.82	96.57	97.14	95.92	92.65	90.47
	Processing time (s)	28.75	29.89	25.33	28.49	26.22	25.74
Medium Neural Network	Acc	98.10	97.90	99.10	98.90	99.00	99.90
	Sensitivity	98.75	97.86	98.57	98.28	97.44	100.00
	Specificity	95.71	97.85	99.71	99.14	99.31	99.84

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
	Youden Index	94.46	95.71	98.29	97.43	96.75	99.84
	Processing time (s)	35.10	50.35	34.90	37.94	34.68	33.67



(a)

(b)

Figure 8. Comparison of (a) performance and (b) processing time of models trained using same number of Night/Visible images

Within Table 5, the accuracy of SVM models spans from 98.40% to 99.00%, whereas medium neural network models exhibit accuracy ranging from 97.90% to 99.90%. Broadly speaking, models trained with CVC-14 night/visible images tend to display lower accuracy compared to those constructed using CVC-14 night/FIR images. This discrepancy can be attributed to the visible camera's images being relatively blurrier than those captured by the infrared camera during nighttime.

In Figure 8(a), the J values for SVM models display a noticeable trend. They rise from 94.82% at the 1:0.25 ratio to 96.57% at the 1:0.5 ratio, reaching their peak at 97.14% for the 1:1 ratio. However, following this peak, the J value consistently declines, reaching 95.92% at the 1:2 ratio, 92.65% at the 1:5 ratio, and finally resting at 90.47% for the 1:10 ratio. This suggests that the optimal ratio for utilizing the CVC-14 night/visible images with the SVM model is the 1:1 ratio.

For medium neural network models, a similar pattern in J values is observed, mirroring the SVM models. They ascend from 94.46% at the 1:0.25 ratio to 98.49% at the 1:1 ratio and then decline until reaching 96.75% at the 1:5 ratio. However, intriguingly, the J value rebounds to 99.84% at the 1:10 ratio. This exceptional value at the 1:10 ratio suggests that, in this scenario, the model may have become extremely cautious and classified nearly all instances as positive to avoid false negatives. Consequently, the most suitable ratio for the medium neural network with this database is also the 1:1 ratio.

Regarding processing time, as illustrated in Figure 8(b), SVM models consistently exhibit processing times ranging from approximately 25.33 seconds to 29.89 seconds. In contrast, medium neural network models display a broader range, spanning from 33.67 seconds to 50.35 seconds. Notably, the processing time for the 1:0.5 ratio is comparatively longer. This can be attributed to the inherent variability in neural network training complexities, particularly when dealing with imbalanced datasets like the 1:0.5 ratio, where positive examples are abundant. In such cases, the network may require additional iterations and time to converge to an optimal solution.

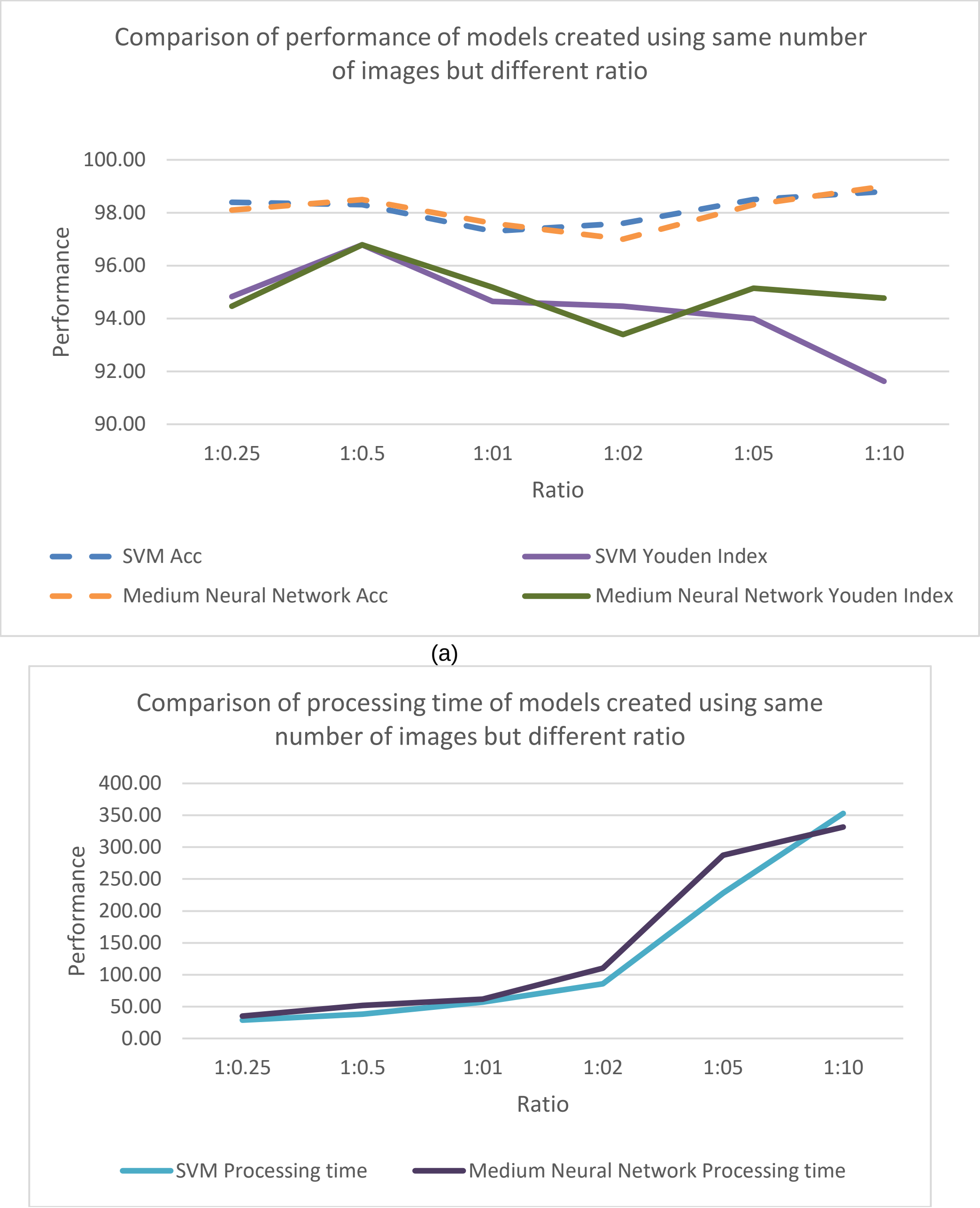
Comparison of the Same Number of Positive Images during Training Stages

In Table 6, the performance of SVM models is highlighted by an accuracy range spanning from 97.30% at the 1:1 ratio to a peak of 98.80% at the 1:10 ratio. Overall, the SVM model consistently maintains a commendably high level of accuracy across all ratios, with its pinnacle achieved at the 1:10 ratio. Conversely, medium neural network models demonstrate accuracy ranging from 97.00% at the 1:2 ratio to an impressive 99.00% at the 1:10 ratio. These findings underscore the medium neural network's capacity for sustaining consistently high accuracy, ultimately reaching its zenith at the 1:10 ratio.

As indicated in Figure 9(a), the Youden Index (J) for both SVM and medium neural network models follows a parallel trajectory. Beginning at the 1:0.25 ratio, it ascends and reaches its zenith at the 1:0.5 ratio. Subsequently, a general decline in the Youden Index is observed for both SVM and medium neural network models as the data ratio increases. This pattern signifies that, with an increase in the ratio, the models adopt a more cautious and balanced approach in their predictions. Consequently, according to the Youden Index, the optimal ratio for developing models for CVC-14 night/visible images is determined to be the 1:0.5 ratio, a choice that holds for both classifiers.

Table 6. Result of the models trained with different positive-negative ratios using the same amount of positive Night/Visible images

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
SVM	Acc	98.40	98.30	97.30	97.60	98.50	98.80
	Sensitivity	99.11	98.21	97.14	96.25	94.82	92.14
	Specificity	95.71	98.57	97.50	98.21	99.18	99.48
	Youden Index	94.82	96.79	94.64	94.46	94.00	91.63
	Processing time (s)	28.75	38.21	56.78	85.83	228.06	352.93
Medium Neural Network	Acc	98.10	98.50	97.60	97.00	98.30	99.00
	Sensitivity	98.75	98.57	97.86	95.71	96.43	95.36
	Specificity	95.71	98.21	97.32	97.68	98.71	99.41
	Youden Index	94.46	96.79	95.18	93.39	95.14	94.77
	Processing time (s)	35.10	51.72	61.57	110.43	287.49	331.59



(a)

(b)

Figure 9. Comparison of (a) performance and (b) processing time of models trained using same number of positive Night/Visible images

Furthermore, it's noteworthy that processing time escalates for both SVM and medium neural network models as the data ratio veers towards greater imbalance, with the 1:10 ratio exhibiting the lengthiest processing times. This trend aligns with the well-established understanding that imbalanced datasets, where one class predominates significantly, tend to necessitate extended training durations, particularly in the case of neural networks.

INRIA Database

Comparison of the Same Number of Images during Training Stage

In Table 7, the performance metrics and processing times for both SVM and medium neural network models across different data ratios are presented. SVM models exhibit accuracy ranging from 97.90% at the 1:1 ratio to 99.00% at the 1:2 and 1:5 ratios. The accuracy generally remains high across all ratios, with some fluctuations. Medium neural network models also maintain relatively high accuracy, with values ranging from 97.80% at the 1:5 ratio to 99.00% at the 1:2 and 1:10 ratios. Like SVM models, medium neural network models demonstrate consistent accuracy performance across the ratios.

Sensitivity for SVM models varies across the ratio. It starts with 99.58% at 1:0.25 ratio and declines to 98.57% at 1:1 ratio. The sensitivity for 1:2 ratio reached 100% and then drops to 95.19% at 1:10 ratios. For medium neural network models, sensitivity ranges from 90.74% at the 1:10 ratio to 100% at the 1:2 ratio. Like SVM models, there is sensitivity to data imbalance, with a drop observed at the 1:10 ratio.

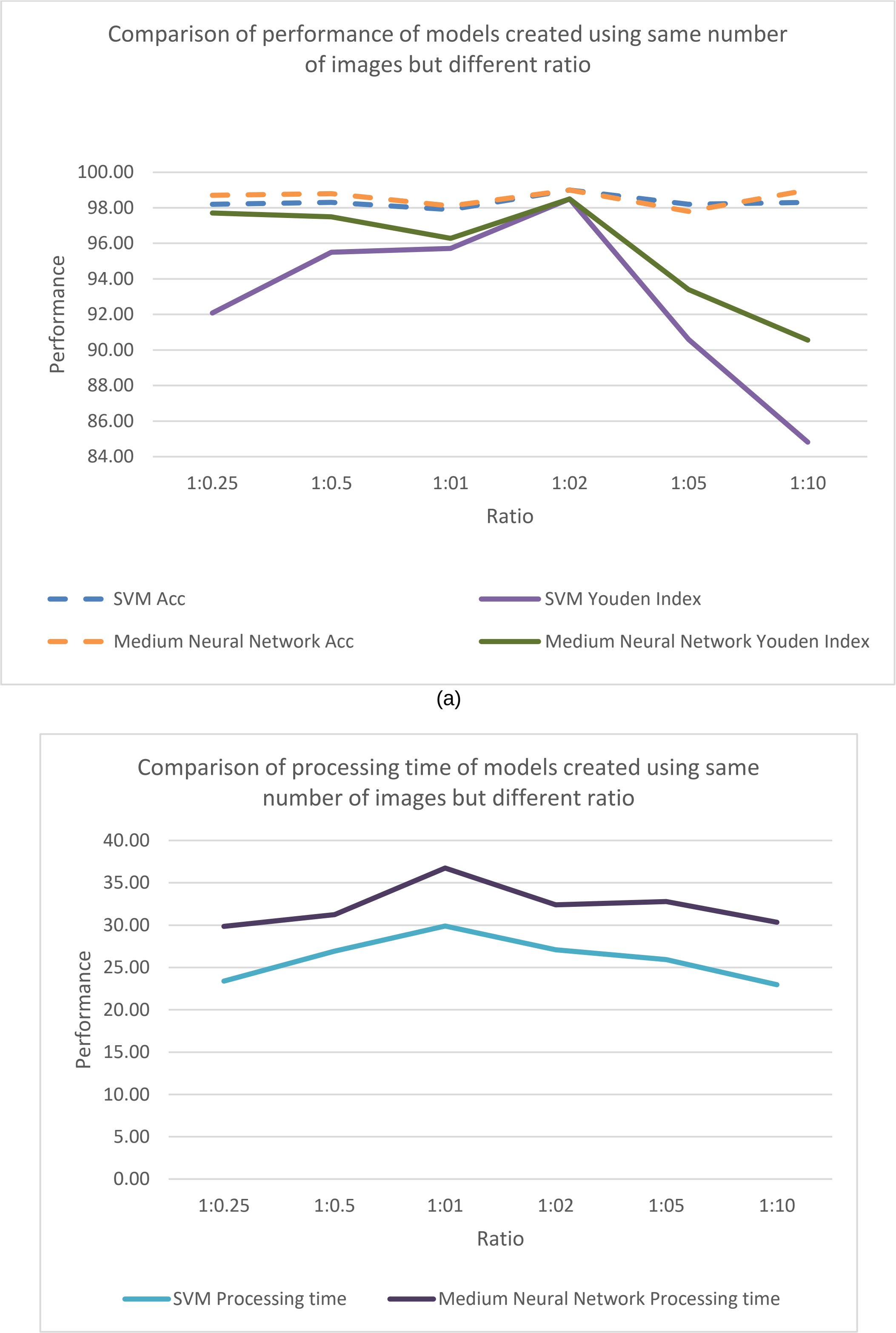
Specificity for SVM models exhibits an increasing trend with data ratios, starting at 92.50% for the 1:0.25 ratio and reaching 99.63% at the 1:10 ratio. This suggests that SVM models become more specific as the data becomes more imbalanced. Medium neural network models show specificities ranging from 97.71% at the 1:1 ratio to 99.82% at the 1:10 ratio. While there is an increasing trend, it's worth noting that specificity remains relatively high across all ratios for medium neural network models.

In Figure 10(a), the Youden Index (J) exhibits parallel trends for both SVM and medium neural network models. The J value reaches its pinnacle at the 1:2 ratios, achieving a robust 98.5%. This signifies the highest discriminatory power observed at these ratios. However, it's noteworthy that there is a noticeable decline in J values at the 1:10 ratio, indicative of diminished discriminatory capability in the presence of significant data imbalance.

Similar to patterns observed in other database models, the processing time for medium neural network models surpasses that of SVM models. Figure 10(b) illustrates this, with processing times for SVM models ranging from 22.96 seconds to 29.90 seconds. Conversely, medium neural network processing times span from 29.86 seconds to 36.75 seconds. This discrepancy in processing times aligns with the convention that neural networks, due to their architectural complexity and training requirements, typically entail longer computational durations compared to SVM models.

Table 7. Result of the models trained with different positive-negative ratios using same number of Night/Visible images

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
SVM	Acc	98.20	98.30	97.90	99.00	98.20	98.30
	Sensitivity	99.58	99.50	98.57	100.00	91.00	85.19
	Specificity	92.50	96.00	97.14	98.50	99.60	99.63
	Youden Index	92.08	95.50	95.71	98.50	90.60	84.82
	Processing time (s)	23.39	26.93	29.90	27.09	25.93	22.96
Medium Neural Network	Acc	98.70	98.80	98.10	99.00	97.80	99.00
	Sensitivity	98.54	99.00	98.57	100.00	95.00	90.74
	Specificity	99.17	98.50	97.71	98.50	98.40	99.82
	Youden Index	97.71	97.50	96.29	98.50	93.40	90.56
	Processing time (s)	29.86	31.24	36.75	32.39	32.80	30.35



(a)

(b)

Figure 10. Comparison of (a) performance and (b) processing time of models trained using the same number of images

Comparison of the Same Number of Positive Images during Training Stages

In Table 8, both SVM and medium neural network models exhibit consistent performance across various data ratios. These models maintain a high level of accuracy, with SVM achieving slightly lower accuracy compared to the medium neural network. Specifically, SVM accuracy ranges from 98.00% to 99.30%, while medium neural network accuracy spans from 98.00% to 99.40%.

Regarding sensitivity, both SVM and medium neural network models demonstrate decreasing trends as the data ratio increases. For SVM models, sensitivity decreases from 99.58% at the 1:0.25 ratio to 93.54% at the 1:10 ratio. On the other hand, medium neural network sensitivity ranges from 98.54% at the 1:0.25 ratio to 95.63% at the 1:10 ratio. This decreasing sensitivity trend signifies that as the data becomes more imbalanced, the models become less sensitive to detecting positive instances.

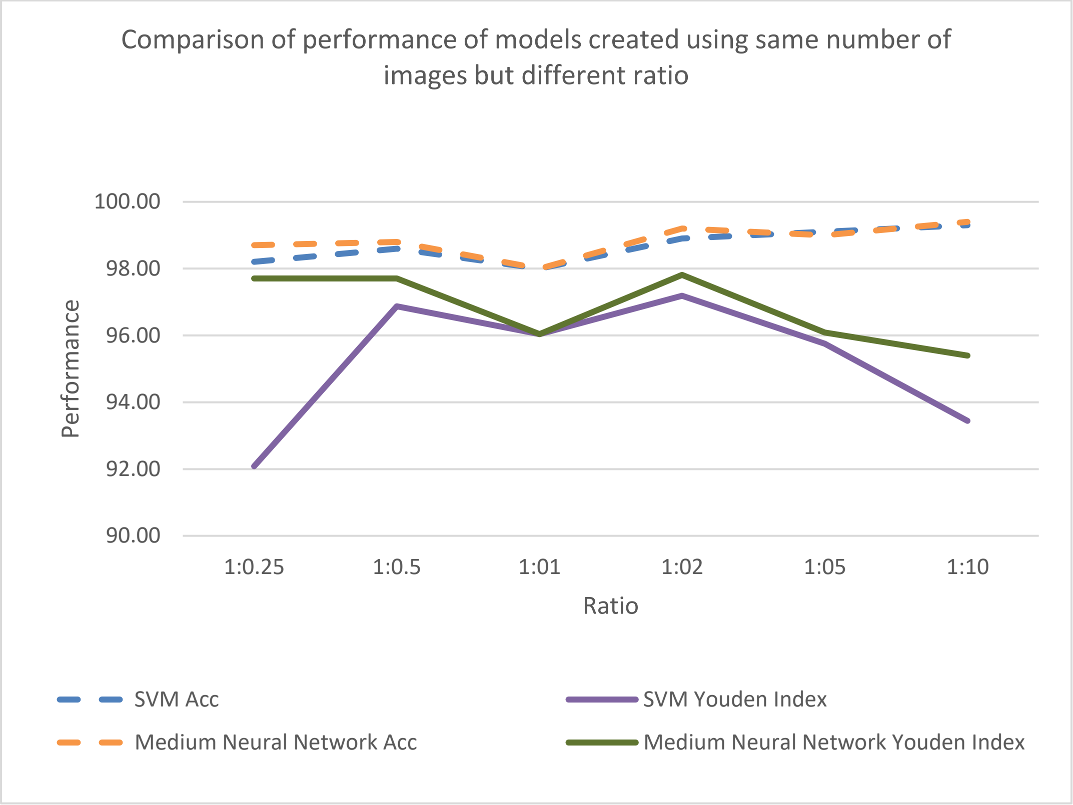
In terms of specificity, both SVM and medium neural network models exhibit an increasing trend as the data ratio increases. SVM specificity ranges from 92.50% at the 1:0.25 ratio to 99.90% at the 1:10 ratio. Medium neural network specificity varies from 99.17% at the 1:0.25 ratio to 99.77% at the 1:10 ratio. This pattern suggests that as the data ratio becomes more imbalanced, the models become more specific in correctly identifying negative instances.

As depicted in Figure 11(a), the J values for SVM models consistently fall below those of the medium neural network across all ratios. Nevertheless, they exhibit a parallel trend, with both achieving their highest J values at the 1:2 ratio. Specifically, the SVM model reaches 97.19%, while the medium neural network model attains 97.81%.

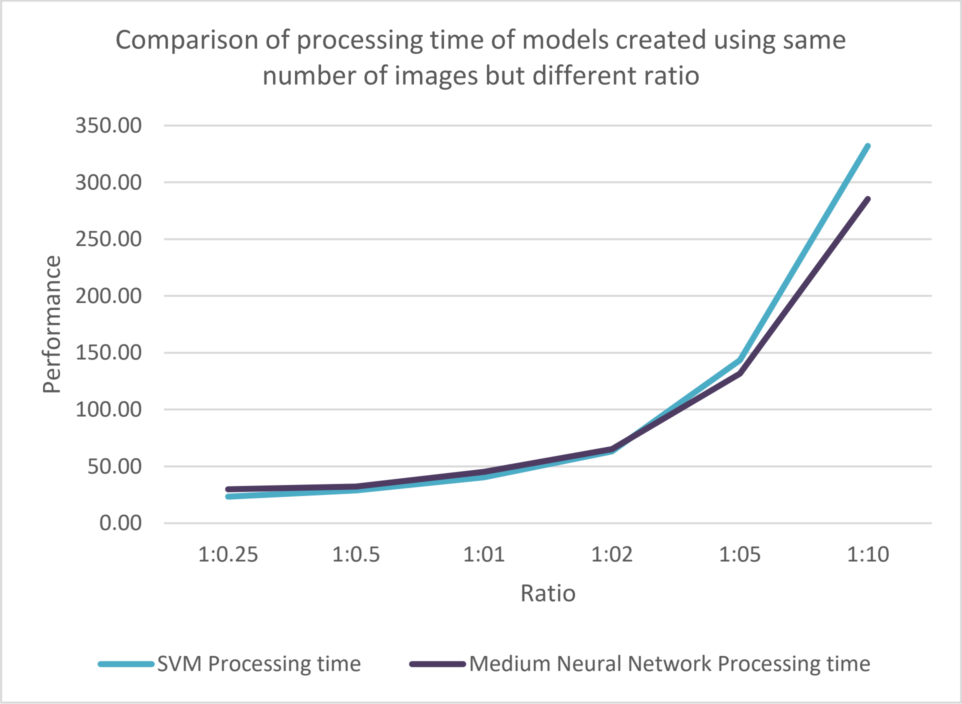
Processing time increases for both SVM and medium neural network models as the data ratio becomes more imbalanced. The 1:10 ratio exhibits the longest processing times for both classifiers. This aligns with the typical observation that highly imbalanced datasets, where one class dominates, can lead to longer training times, especially for neural network models due to their complexity and iterative training processes.

Table 8. Result of the models trained with different positive-negative ratios using the same amount of positive Night/Visible images

Classifier	Ratio	1:0.25	1:0.5	1:01	1:02	1:05	1:10
SVM	Acc	98.20	98.60	98.00	98.90	99.10	99.30
	Sensitivity	99.58	98.96	98.13	97.71	96.04	93.54
	Specificity	92.50	97.92	97.92	99.48	99.71	99.90
	Youden Index	92.08	96.88	96.04	97.19	95.75	93.44
	Processing time	23.39	28.87	40.38	63.37	143.59	332.12
Medium Neural Network	Acc	98.70	98.80	98.00	99.20	99.00	99.40
	Sensitivity	98.54	98.54	97.71	98.13	96.67	95.63
	Specificity	99.17	99.17	98.33	99.69	99.42	99.77
	Youden Index	97.71	97.71	96.04	97.81	96.08	95.40
	Processing time	29.86	32.37	45.20	65.10	131.61	285.41



(a)



(b)

Figure 11. Comparison of (a) performance and (b) processing time of models trained using the same number of positive images

Conclusions

The primary objective of this study is to determine the optimal ratio of positive to negative images when training HOG-SVM and HOG-CNN models for pedestrian detection. This investigation explores two distinct cases and encompasses six different image ratios sourced from the CVC-14 night/FIR database, CVC-14 night/visible database, and INRIA database to build and evaluate the detection models. The experimental findings emphasize that the choice of positive and negative image ratios significantly influences the accuracy, sensitivity, and specificity of the detection models. In general, across all databases, both classifiers consistently exhibit remarkable accuracy across various data ratios. Sensitivity and specificity display an inverse relationship as the data ratio increases. Based on the Youden Index, the study reveals that, on average, the medium neural network model achieves slightly higher values than the SVM model. Consequently, the recommended ratios for constructing human detection models using both classifiers are 1:0.5 and 1:2.

Notably, concerning processing time, the medium neural network consistently demands more time than the SVM model across all databases. This discrepancy in processing time can be attributed to several factors, including model complexity and batch size. Neural networks, particularly medium-sized ones, typically feature a greater number of parameters and layers compared to linear SVMs. The heightened model complexity necessitates more computational resources during both training and inference, contributing to extended processing times. Additionally, neural networks often benefit from larger batch sizes during training, which further amplifies processing time. Conversely, SVMs generally operate with smaller subsets of data during each iteration.

The findings also illuminate a tendency towards overfitting at both the 1:1 and 1:10 data ratios. When the positive and negative classes are perfectly balanced, there exists a heightened risk of overfitting, especially for intricate models such as neural networks. Overfitting transpires when a model overlearns noise in the data rather than capturing the underlying patterns, resulting in diminished generalization performance on unseen data. At the 1:10 ratio, where negative instances significantly outnumber positive ones, the model may exhibit an inclination to classify nearly all instances as positive to mitigate the potential for false negatives.

However, it is important to acknowledge that the study's focus on specific datasets or scenarios may limit the seamless generalization of the identified optimal ratio to diverse environments. Real-world scenarios, characterized by variations in complexity, lighting conditions, and pedestrian densities, pose challenges not fully addressed in the study. Additionally, the investigation into varying ratios may not thoroughly explore the severity of imbalance, particularly in extreme cases. Insights into how extreme imbalances impact model performance and whether specialized handling techniques are needed remain areas for further exploration.

To enhance the models' performance, several avenues for future research are recommended based on the findings of this study. Firstly, it is advisable to consider the use of class weighting during training to address the challenge posed by class imbalance. This approach can be especially beneficial for ratios that are susceptible to overfitting, such as the 1:1 ratio. Secondly, there is potential to harness hardware accelerators like GPUs or TPUs for both model training and inference, which can substantially reduce processing time.

Further investigation into the intricate relationship between model complexity and the optimal ratio is warranted. Complex models may exhibit unique preferences for positive to negative ratios, offering valuable insights into model architecture design. Instead of pursuing a universal optimal ratio, future research can shift its focus towards exploring the utilization of multiple ratios and evaluating their impacts on model performance. Gaining insights into the trade-offs in various scenarios will empower researchers to make informed decisions, allowing them to select ratios that are specifically tailored to meet the requirements of diverse applications..

Additionally, further investigation into different feature extraction techniques should be conducted. This includes exploring various HOG variations and more advanced feature extraction methods to capture more comprehensive and informative data from the images. Lastly, it is essential to expand the experimentation to real-time applications to assess the models' performance in practical, real-world scenarios.

These proposed directions can contribute to the continued advancement of pedestrian detection models, ultimately enhancing their accuracy, speed, and applicability in diverse settings.

Conflicts of Interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

Acknowledgment

The author would like to express their appreciation to Takasago Thermal Engineering Co, Ltd., Japan (R.K130000.7343.4B792) for providing financial support for this work.

References

[1] Tarek, B. and R. Liu. (2013). *Study and performance analysis of motion detection algorithms*. 355-358.

[2] Liu, J., *et al*. (2018). Multi-target intense human motion analysis and detection using channel state information*. Sensors, 18*(10), 3379.

[3] Yun, J. and S. S. Lee. (2014). Human movement detection and identification using pyroelectric infrared sensors. *Sensors (Basel),* 14(5), 8057-81.

[4] Balogh, Z., M. Magdin, and G. Molnár. (2019). Motion detection and face recognition using raspberry pi, as a part of, the internet of things. *Acta Polytechnica Hungarica, 16*(3), 167-185.

[5] Sumit, S.S., D. Rambli, and S. Mirjalili. (2021). Vision-based human detection techniques: A descriptive review. *IEEE Access. 9*, 42724-42761.

[6] Vrigkas, M., C. Nikou, and I. A. Kakadiaris, (2015). A review of human activity recognition methods*. Frontiers in Robotics and AI, 2*.

[7] Cheng, W.-H., *et al*. (2021). Fashion meets computer vision: A survey*. ACM Computing Surveys (CSUR), 54*(4), 1-41.

[8] Sumit, S. S., D. R. A. Rambli, and S. Mirjalili. (2021). Vision-based human detection techniques: A descriptive review. *IEEE Access*, *9*, 42724-42761*.*

[9] Patwary, M. J. A., S. Parvin, and S. Akter. (2015). Significant HOG-histogram of oriented gradient feature selection for human detection. *International Journal of Computer Applications, 132*, 20-24*.*

[10] Cheung, Y.m. and J. Deng. (2014). Ultra local binary pattern for image texture analysis. *Proceedings 2014 IEEE International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*.

[11] Shorten, C. and T.M. Khoshgoftaar. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data, 6*(1), 60.

[12] Thabtah, F., *et al*. (2020). Data imbalance in classification: Experimental evaluation. *Information Sciences*, *513*, 429-441.

[13] Dalal, N. and B. Triggs. (2005). Histograms of oriented gradients for human detection. *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*. IEEE.

[14] Ojala, T., M. Pietikäinen, and T. Mäenpää. (2000). Gray scale and rotation invariant texture classification with local binary patterns. *European conference on computer vision*. Springer.

[15] Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the seventh IEEE international conference on computer vision*. IEEE.

[16] Zhang, S., C. Bauckhage, and A. B. Cremers. (2014). Informed haar-like features improve pedestrian detection. *Proceedings of the IEEE Conference on computer vision and pattern recognition*.

[17] Tan, X. and B. Triggs. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE transactions on image processing*, *19*(6), 1635-1650.

[18] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(9), 1627-1645.

[19] P. Viola and M. Jones (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Kauai, HI, USA, I-I.

[20] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, *110*(3), 346-359.

[21] Zhu, Q., *et al*. (2006). Fast human detection using a cascade of histograms of oriented gradients. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR'06), *2*, 1491-1498.

[22] He, N., J. Cao, and L. Song. (2008). Scale space histogram of oriented gradients for human detection. *2008 International Symposium on Information Science and Engineering*.

[23] Watanabe, T., S. Ito, and K. Yokoi. (2009). Co-occurrence histograms of oriented gradients for pedestrian detection. *Pacific-Rim Symposium on Image and Video Technology*. Springer.

[24] Hiromoto, M. and R. Miyamoto. (2009). Cascade classifier using divided CoHOG features for rapid pedestrian detection. *International Conference on Computer Vision Systems ICVS 2009*, 53-62.

[25] Quintero, R., *et al*. (2017). Pedestrian intention recognition by means of a hidden Markov model and body language. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE.

[26] Jalal, A., S. Kamal, and D. Kim. (2016). Human depth sensors-based activity recognition using spatiotemporal features and hidden markov model for smart environments*. Journal of Computer Networks and Communications,* 2016.

[27] Freund, Y. and R. E. Schapire. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119-139.

[28] Hyeranbyun and w. Seong (2011). A survey on pattern recognition applications of support vector machines. *International Journal of Pattern Recognition and Artificial Intelligence*, *17.*

[29] Sanchez-Matilla, R., F. Poiesi, and A. Cavallaro. (2016). Online multi-target tracking with strong and weak detections. *Computer Vision – ECCV 2016 Workshops.* Cham: Springer International Publishing.

[30] Lipetski, Y. and O. Sidla. (2017). A combined HOG and deep convolution network cascade for pedestrian detection*. Electronic Imaging*, *2017*, 11-17.

[31] Zhang, J., J. Cao, and B. Mao. (2016). Moving object detection based on non-parametric methods and frame difference for traceability video analysis. *Procedia Computer Science, 91*, 995-1000.

[32] Rui, T., *et al*. (2017). Pedestrian detection based on multi-convolutional features by feature maps pruning. *Multimedia Tools and Applications*, *76*(23), 25079-25089.

[33] Chen, N., W.-N. Chen, and J. Zhang. (2015). Fast detection of human using differential evolution*. Signal Processing, 110*, 155-163.

[34] Bahri, H., *et al*. (2020). Real-time moving human detection using HOG and Fourier descriptor based on CUDA Implementation*. Journal of Real-Time Image Processing*, *17*.

[35] González, A., *et al*. (2016). Pedestrian detection at day/night time with visible and FIR cameras: A comparison. *Sensors*, *16*(6), 820.

[36] Trevethan, R. (2017). Sensitivity, specificity, and predictive values: Foundations, pliabilities, and pitfalls in research and practice*. Frontiers in Public Health*, *5*(307).

[37] Fluss, R., D. Faraggi, and B. Reiser. (2005). Estimation of the Youden Index and its associated cutoff point. *Biometrical Journal: Journal of Mathematical Methods in Biosciences*, *47*(4), 458-472.