Detection of vague object signatures on deep learning surveillance devices

I Ketut Swardika, Putri Alit Widyastuti Santiary
Department of Electrical Engineering, State Polytechnic of Bali, Bali, Indonesia

Article Info

Article history:

Received Dec 18, 2023
Revised Feb 26, 2024
Accepted Mar 8, 2024

Keywords:

Deep learning
Near-infrared camera
Night person
Object detection
Surveillance device

ABSTRACT

The deep learning of object detection has become a breakthrough in recent years. Many papers demonstrated that this method records significant reliability results. However, the question arises whether objects that were successfully detected are initially conditioned clear in daylight. The object being detected is in the form of a photographic product that has numerous problems. It can be distant or have low-contrast so that their signatures are challenging to recognize, especially detection of persons in surveillance systems for dark-environments. This paper contributes to proving the deep learning method capable of detecting night-person (NP) with high precision and recall in the dark without image enhancement, by using ordinary cameras which operate on day-night or visible-near infrared spectrum runs on embedded systems. For that, an infrared-cut filter mechanical shutter is designed to block for the day or deliver infrared light for the night. The NP signatures are illuminated by an external infrared light source, providing three-channel high-resolution images. The distance of a NP from the camera becomes a decisive successful detection. The external infrared light source makes objects under or overexposed affecting the object being recognized. The validation with thoroughly new data of the NP constantly provides high precision and recall.

Corresponding Author:

I Ketut Swardika
Department of Electrical Engineering, State Polytechnic of Bali
Kampus Bukit Jimbaran, Bali, Indonesia
Email: swardika@pnb.ac.id

1. INTRODUCTION

Computer-assisted visual intelligence is the most active research topic nowadays. Vision from a camera is the only way an artificial intelligent device sees the world of the environment by the light. Although the other way, a synthetic aperture radar (SAR) [1]–[3] or light detection and ranging (LiDAR) [4]–[6] is able to imagine the world of the environment through radio or light backscattering, however not yet widely applicable. A camera is a passive sensor, that can see the objects from the reflection of the light that is illuminated surface in the visible (VIS) light spectrum 0.4-0.75 µm wavelength. In the light spectrum, the longer wavelength beside VIS is infrared light. The infrared wavelength spectrum covers from 0.75 to 1000 microns (µm) and is divided into several bands i.e., near, mid, shot, long, and far infrared. With their wavelengths near-infrared (NIR) 0.75–1.4, short-wavelength infrared (SWIR) 1.4–3, mid-wave infrared (MWIR) 3–8, long-wave infrared (LWIR) 8–15, and far infrared (FIR) 15–1000 in micron [7]–[9].

A camera sensor also operates in the NIR band as a night-vision camera and in the LWIR band as a thermal-imaging camera when there is insufficient light or dark environment. These two types of sensors have different mechanisms and should not be confused [10], [11]. The NIR camera is an active sensor, which means the camera is equipped with an infrared light source to illuminate an area of interest and capture back reflection of infrared energy, interpreted to generate an image. The NIR camera produces sharp-quality imagery, making objects VIS to the human eye when dark. Because equipped by an infrared light source usually an LED, it suffers within working range and cannot remove obstructions in front of an object [10], [12]. The NIR image has three channels like VIS RGB, but wider sensitivity. Rapid application in the vision industry made the NIR camera low-cost, making it an alternative to VIS imaging on a robust and practical identification system.

The LWIR camera is a passive sensor. This means the object of interest has its own infrared energy to emit as electromagnetic waves to the environment. Some of that infrared energy is captured by the LWIR camera and interpreted to generate an image. The infrared energy source comes from the heat of the object's body, providing thermal information on the self-radiation of an object [13], [14]. These heat signatures are usually cold as black and hot as white displayed on the image. This camera covers wider distances and is not affected by smoke, haze, fog, dust, or oncoming headlights. For comparison, the human body's temperature at 310 deg K (36.85 deg C) has the peak wavelength of black-body radiation at 9.35 μm from Wien's displacement law [15], [16]. The LWIR camera produces coarse quality imagery, heavy noise, and low resolution, details in visual of objects are lost, and only the outlines are conserved due to the lack of information (one channel gray-scale image) and sensitivity to temperature changes in the environment, makes it is vulnerable to warm cold air. This disadvantage makes it challenging for imaging identification systems. Results can be misclassified by the environmental information and suppress the detection accuracy [11].

With computer-assisted visual intelligence, automatic surveillance, and monitoring systems become a must-have installed in private and public areas, as a major concern to security and law enforcement of human and property [17]. This allows data acquired by surveillance cameras to be automatically processed without continuous attention from operators [18], [19]. Nowadays, automatic surveillance systems have reached a level of maturity, embedded with artificial intelligence features on practical applications. Although deep learning using convolutional neural networks (CNN) achieved significant breakthroughs in object detection [20], [21] most results are efforts on VIS images and avoid illumination difficulties i.e., lowlight to absolutely dark environments [22]. Most research on lowlight VIS images commonly addresses image enhancement problems that utilize massive resources or thermal imaging surveillance that demands expensive hardware [23]–[25]. Meanwhile, relatable object detection is given less attention.

For a better understanding and further development of CNN object detection in the dark environment, this paper contributes to this field forward. First, present NIR image datasets of persons in the dark environment completed with ground truth annotation. This is crucial because available datasets that publicly specifically provide NIR images for object focus are occasional. Further challenges arise for data annotation because it is difficult to manage the volume of data. Second, presents an object-focused analysis of NIR images and their differences from VIS images for a better understanding. Finally, it presents the results of object detection using state-of-the-art algorithms and learned features.

2. METHOD

This research requires a dark room such as an indoor corridor or alley where outside light cannot penetrate. The dark room and camera sketch up as seen in Figure 1. The dark room is long enough and narrow enough to avoid stray infrared reflections or sources, and all the stuff inside the room is temporarily removed. In Figure 1, *d* is a distance variable between the night-person (NP) which acts as a thief with the camera. The camera setup consists of a base camera rotator, NIR camera itself, built-in infrared LED, built-in light sensor, passive infrared (PIR) sensor [26], [27] array, and additional 3-watt infrared flashlight used to strengthen the distance of observation. The infrared sources will illuminate objects in front of the camera with distance *d* as the analysis variable. Light-sensor is a light-dependent resistance (LDR) active signal when in dark or low-light conditions. Insert figure in Figure 1 is the construction of a PIR sensor array. PIR sensor observes infrared energy emissions as present in the human body. The PIR array sensor consists of five PIR sensors that are arranged with 15 deg angle observation. For better sensing, PIR is placed inside an aluminum tube, covered by aluminum foil. PIR array sensor brings the active signal to the rotator as a base of the camera body to maneuver, resulting in the presence of a person's body center inside the camera frame.

A single board computer (SBC) Raspberry Pi 4 8 Gb RAM (RPi) is used for object detection processing and microcontroller for human detector and camera base rotator [28]–[30]. The hardware wiring diagram is seen in Figure 2. A camera package with a light sensor and built-in infrared LED bond to the camera serial interface (CSI) of RPi. The rotator is a 2-phase bipolar stepper motor (NEMA) with 0.9 deg per step driven by a 4-channel L298N motor driver. The 4-channel input (in1-in4) connects to 4 of RPi GPIO i.e., pin29-pin35. The rotor was initially set up at zero deg, the center of the camera base by lever-switch pin11 of RPi GPIO. The rotator will turn clockwise (cw) or counterclockwise (ccw) according to the PIR array signal. The PIR array is HC-SR505 mini type connected to pin36-pin40 of RPi GPIO.

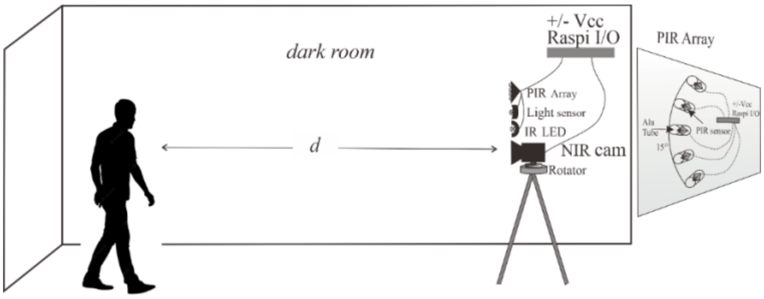


Figure 1. Research setup

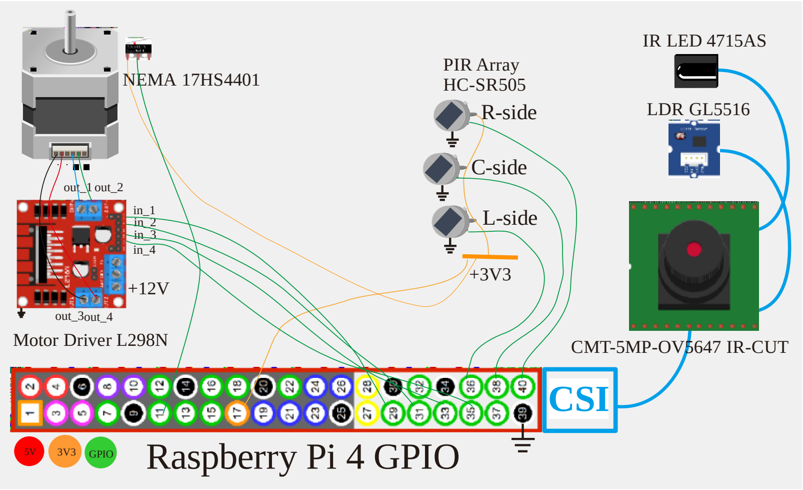


Figure 2. Hardware wiring diagram

The camera uses a 5 MP (ov5647) with an infrared-cut filter feature. The infrared-cut filter operates with a mechanical shutter to block infrared light for the day or deliver infrared light for the night or in low light conditions. This provides a true color high-quality image regardless of day or night. The infrared-cut filter mounts between lens and image-sensors as shown in Figure 3. The infrared -cut filter mechanically controlled onboard by a motor or an electromagnet coil that pushes or pulls the filter based on sense by light-sensor and turns on a built-in infrared LED. The RPi runs Debian Bullseye 64-bit OS with remote access via a virtual network computing (VNC) server and client. For a better frame per second (FPS) of object detection, an edge tensor processing unit (TPU) accelerator is attached to RPi. The mechanical camera setup in Figure 2 is operated by Python programming which accesses the RPi GPIO through the RPi-GPIO library. The RPi camera module was initiated using the picamera2 library. For NP object detection, this research used the TensorFlow2 (TF2) object detection application programming interface (API) [16].

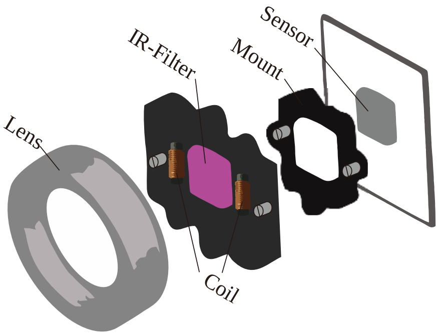
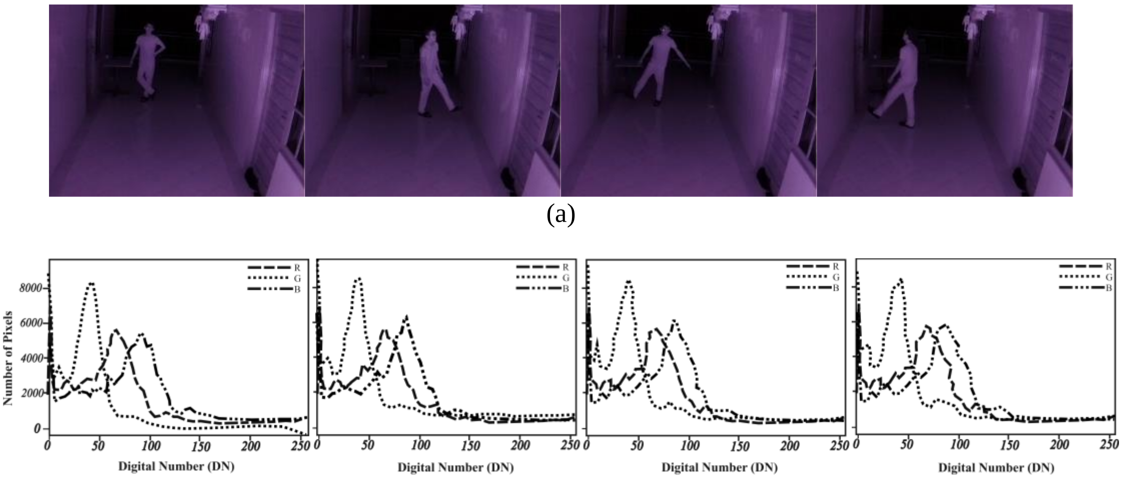


Figure 3. Infrared-cut mechanism

2.1. Night-person datasets collection

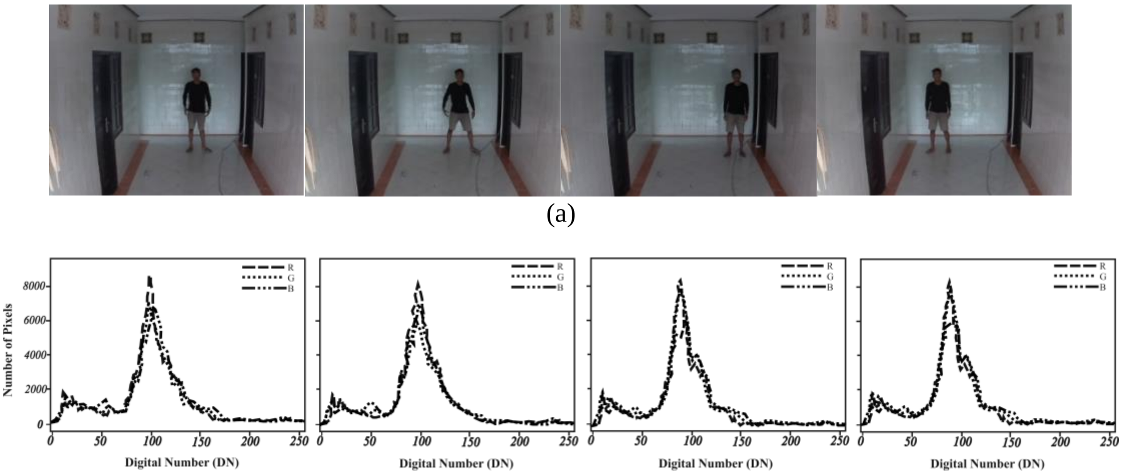
The first research step is collecting of images NP datasets using the research setup in Figure 1. The NP stand-in male with attribute wearing a facemask, hat, jacket, or hoody as one class object. Walk randomly to the camera with attention to variable *d* as distance. For simplification, they act over a line mark of 7 m and 6 m. The NP images are captured automatically using Python CV2 programming with dimensions 640 by 480 pixels. The NP images focused only on object structure and omitted detail on the face. Total images NP datasets collected about 800 images. For an early preview, the NP image datasets with their BGR histogram are present in Figures 4(a) and 4(b) in this section, further analyses and discussion will be presented in the next section. For better understanding, the NP image is also compared with VIS day_person (DP) images with their BGR histogram in Figures 5(a) and 5(b).



(a)

(b)

Figure 4. The NP image datasets, (a) NP images in the dark room and (b) their histogram BGR channel respectively



(a)

(b)

Figure 5. The DP image datasets, (a) DP images on the day room and (b) their histogram BGR channel respectively

2.2. Object detection

Object detection is a computer vision (CV) task concerned with automatically finding semantic objects in an image, through a training process with a bunch of images and respective annotations. Before that, CV is a hard subject to get running, strong understanding of the underlying computer infrastructure and deep knowledge of machine learning to achieve a minimally acceptable performance. Today, with the breakthroughs of deep learning, object detectors achieve state-of-the-art accuracy and impressive real-time FPS rates [14], [16]. With the TF2 object detection API framework, it's now more convenient for writing complex models like object detectors with only concerns about hyperparameters tunning to achieve the desired performance [10].

The model of object detection is a combination of two tasks i.e., classification of the object label and regression of the bound-box coordinates as seen in Figure 6. This means that the model has two output branches. The streamlining of the model consists of the input layer, feature extraction or hidden layer, and output layer. The common structure of the feature extractor is composed of a sequence of convolution layers (CNNs, ReLU activation, and pooling) [10]. Features are something like building blocks of images i.e., low-level regularities in the training data. The extracted features are fed into the adapter layer, which is basically a flattened layer and ends with two output branches (fully connected layers) i.e., the classifier and the regressor. Both classifier and regressor have a similar architecture, except at the end the classifier has only one output i.e., class label while the regressor has a four-unit output i.e., bounding box coordinates. During the training process, each output branch will be specialized to its respective task.

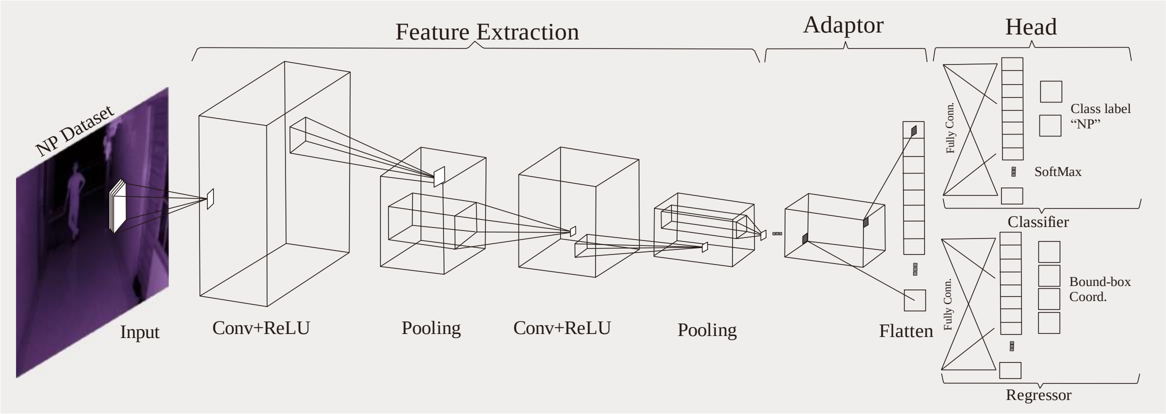


Figure 6. NP object detection architecture

In order to learn, the network provided consistent knowledge about objects. This process is known as ground truth or annotation images as seen in Figure 7. With this process, set pixels of object structure i.e., the NP class object segmented from others set pixels. The annotation information consists of the source file, class name, and geometric segmentation. The annotation file in XML for pattern analysis, StatistiCAL modeling, and computational learning (PASCAL) visual object classes (VOC) format is separated from the image file [1]. Produce a total image dataset of about 800 images (JPEG) and their 800 annotation-file (XML) respectively. Data preparation set before mounting to COLAB drive, divided into three folders i.e., randomly selects 80 percent for train, 10 percent test for evaluation, and the remaining 10 percent for validation to ensure the training process is not overfitting [14].

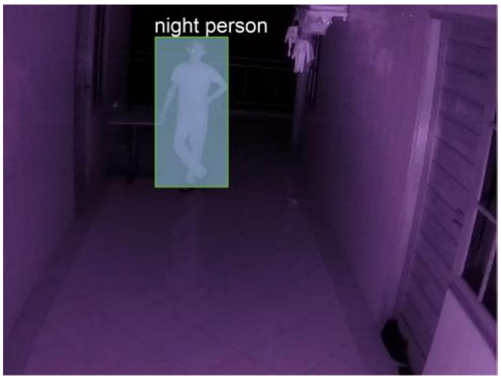


Figure 7. NP annotation

As the target model will be running on an SBC RPi, the model output must be light enough. Therefore, the model output must be exported into the lite version of TF2 (TFlite). One of the lite model versions that is developed specifically for SBC and IoT devices is the efficient detector lite model. This model assures running well with reasonable accuracy on RPi. The training process is run on Colab with TFlite model maker library installed inside Conda virtual environment. For train model evaluation, simply use COCO metric evaluator format provided by cocotools library [23].

As seen in Figure 6, the model output has two branches, the classifier and regressor. Both outputs are greatly different, therefore at least two metric evaluators are needed. Since the objective of object detection is to correctly classify the objects and where those objects are in the image, image classification metrics such as precision and recall cannot be simply used. Therefore, the metric evaluator offers both classification and geometric consideration. First, for geometric consideration use intersection over union (IoU) pixels. Its ratio between overlap and union of ground-truth bound-box pixels and predicted bound-box pixels, as seen in (1). The number will be zero to one, where one meaning exactly matches both bond boxes. At least an IoU threshold of 0.5 is typically considered. Second, for classifier offers a confidence score (CS) as seen in (2), which is the probability of the image being detected correctly by the network and is given as a percentage [14].

$$IoU = \frac{\sum_{px}(G \cap P)}{\sum_{px}(G \cup P)} = \frac{\sum_{px}(\min_{xy}(G,P) - \max_{xy}(G,P))}{\sum_{px}(G + P - \min_{xy}(G,P) - \max_{xy}(G,P))} \tag{1}$$

Where G and P are ground truth and predicted pixels. While min/max G and P is the minimum and maximum coordinates of G, P, or width and height of intersection areas.

$$CS = ClassProbablity.100\% \tag{2}$$

To improve or benchmark model results, sometimes need a single-number metric. However, cannot simplify multiply IoU and CS. Because object detection considers a qualitative number of success or failure predictions on sample test datasets. Therefore, they are divided into true-positive (TP), when actual (true) and predict class agrees (positive), otherwise true-negative (TN). Errors will come when the actual (false) and predicted class agree (positive) false-positive (FP) or the predicted class disagrees (negative) false-negative (FN). All bounds will be summaries from their confusion matrix. When measuring the precision or accuracy of the model in classifying samples as positive, bring up precision as in (3). And how sensitive or recall of model in classifying samples as the right prediction out of all predictions, present in (4) [31].

$$Precision = TP/(TP + FP) \tag{3}$$

$$Recall = TP/(TP + FN) \tag{4}$$

Both metrics should reach maximum, otherwise, there is a trade-off between them which causes issues in the model's performance. In practice, to get the best compromise between these metrics by setting a threshold on the precision-recall curve. For each setting threshold respects their precision after interpolation (P_{interp}), by average all precision (AP) across all unique recall (R) levels ends AP as seen in (5) [32].

$$AP = \sum_{i=1}^{n-1} (R_{i+1} - R_i)\, P_{interp}(R_{i+1}) \tag{5}$$

If the model detects more than one class, (5) becomes mean average precision (mAP) across all classes (K) as seen in (6). Within the training process, model NP will be evaluated with metric mAP [23].

$$mAP = \frac{\sum_{i=1}^{K} AP_i}{K} \tag{6}$$

Analysis will be conducted from training to implementation results. In the training process with fixed model selection, conversion model format from tensor flow to lite version of tensor flow and edge TPU format affects precision and performance. The amount of training datasets not only affects the time-step to the longer training process but possibility wider of validation loss. These results will be analyzed using a statistical comparison method. In the implementation, as an NIR camera suffers with the working range the NP detection performances will be associated with object distance. The mAP result will be obtained from real-time detection of NP wearing various attributes.

3. RESULTS AND DISCUSSION

The NP that was captured by the NIR camera as seen in Figure 4(a) is a three-channel BGR image. Images look dark with predominantly purple. The NP pixel's object structure is clearly distinguished and brighter compared with the darkroom. The three-channel BGR histogram of NP in Figure 4(b) is expected to be identical or in grayscale but not, all channels can be distinguished. The green channel is more dominant. Like dark images, histograms assemble to the left toward zero. The arithmetic mean of all pixels is about 53 and the median statistic is about 51. Since NP is captured by an active NIR camera in the darkroom which camera sensor absorbs reflected infrared energy, and the NP pixel's structure is clearly obtained even person wears a dark outfit. Comparison with DP images in Figure 5(a) shows a person standing in a predominantly white-paint room. A person wearing a dark shirt clearly sees contrast with the room color. The DP histogram in Figure 5(b) shows channels that distribute and accumulate in the center of the histogram. This is due to DP images dominated by white color. With no light, DP captured by a passive VIS camera in the darkroom results in nothing only dark images.

Kristo *et al.* [33] review images resulting from thermal imaging (LWIR) cameras which provide much less detail and lower resolution. The pixel body of NP tends to be blurred and cannot determine the boundary surrounding the background. Dai *et al.* [34] provide a comprehensive comparison of NP capture by VIS, NIR, SWIR, and LWIR cameras. And state that longer operational wavelength causes lower resolution and detail but is more expensive.

3.1. Mean average precision network format evolution

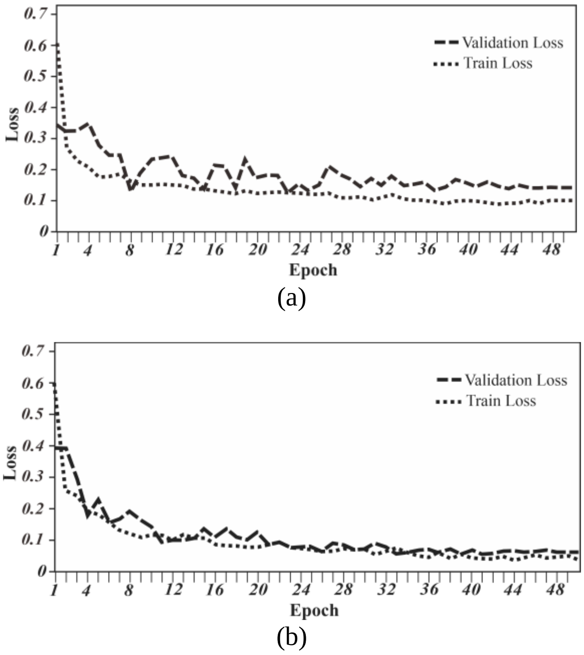
The NP dataset had been trained on Google Colaboratory with GPU hardware accelerator. Within a total of 800 images and 50 epochs, the train process took 69.65 minutes to finish. During the training process, logs were saved for further analysis. Since the model will run on SBC RPi, the network format evolution is presented, and the results are shown in Table 1. The mAP metric is a primary challenge metric, its computed within 10-step thresholds of IoU starting from 0.50 to 0.95 [16]. This is a standard metric to evaluate the performance of object detectors, however, there are variations among the metrics shown in Table 1. The mAP50 or mAP70 is mAP computed with single thresholds of IoU. Another variant of mAP computes across scales with large, medium, and small areas. The mean average recall (mAR) is another variant that uses recall rather than precision [23]. The mAP_/NP overall high over 0.85, within network format conversion to edge TPU a slick reduces to 0.84. This network format evolution reduces the workload to 38 percent as seen from network file size (MB). These mAP results are comparable with the related study in [16], where obtained mAP around 0.70-0.80. Additionally, the smaller network size improves the precision of real-time detection [14].

Table 1. Metrics network format comparison

Metrics	TF	TF lite	Edge TPU
mAP	0.87263596	0.86358064	0.8425685
mAP50	1.0	1.0	1.0
mAP75	1.0	1.0	1.0
mAP_/NP	0.87263596	0.86358064	0. 8425685
mAPl :	0.8726906	0.86359024	0. 8425685
mAPm :	-1.0	-1.0	-1.0
mAPs :	-1.0	-1.0	-1.0
mARl :	0.91625	0.88375	0.84565
mARm :	-1.0	-1.0	-1.0
mARmaxl	0.88502	0.88465	0.84375
mARmaxl0	0.91625	0.88375	0.84265
mARmaxl00	0.91625	0.88375	0. 84265
mARs	-1.0	-1.0	-1.0
Size(MB)	7.2	5.8	4.4

3.2. Loss of train and validation

In the training process to ensure the training process runs properly, loss is used to measure quantities of the error produced by the network. High loss means the network produces erroneous output. Two losses are mostly used to assess network performances i.e., train and validation loss. The training loss indicates training data fits by the network, while validation loss shows how the network fits on new data. Figure 8 shows the network learning curve by both losses. To omit overfit or long-period training, full dataset (within 800 images) and halt dataset to be tested. Results shown in Figure 8(a) with a full dataset, the network tends to be overfit. In Figure 8(b) gap between losses becomes small, indicating that the network optimally fits and can generalize on new data. These results were confirmed by an investigator in [14], who stated that deeper networks, heavy datasets, and longer training processes cause overfitting.



(a)

(b)

Figure 8. Comparison of train and validation loss for (a) full datasets and (b) halt datasets

3.3. Night-person detection

The real-time NP detection also considers of distance NP from the camera, as known the NIR camera heavily depends on an active infrared light source. Figure 9 shows real-time NP detection at various distances from the camera. NP distance gradually longer from the upper-left to the bottom-right image. The blue bounding box labeling with CS indicates NP has been detected. This figure is also printed of FPS detection with an average of 6 FPS beside time stamps. The real-time NP detection has been tested 10 times for each distance and results in good performance. In general, the real-time NP was successfully detected with a high CS above 0.7 for various distances between 1 m to 6 m. Kristo *et al.* [33] carefully determine the distance of thermal imaging cameras to successfully detect humans in various weather conditions. In foggy or rainy weather conditions, for successful human detection, the distance is close enough. Dai *et al.* [34] use a strong infrared light source installed on a car to get enough distance (about 20 m) for pedestrian detection. These results show that distance is crucial in nighttime object detection.

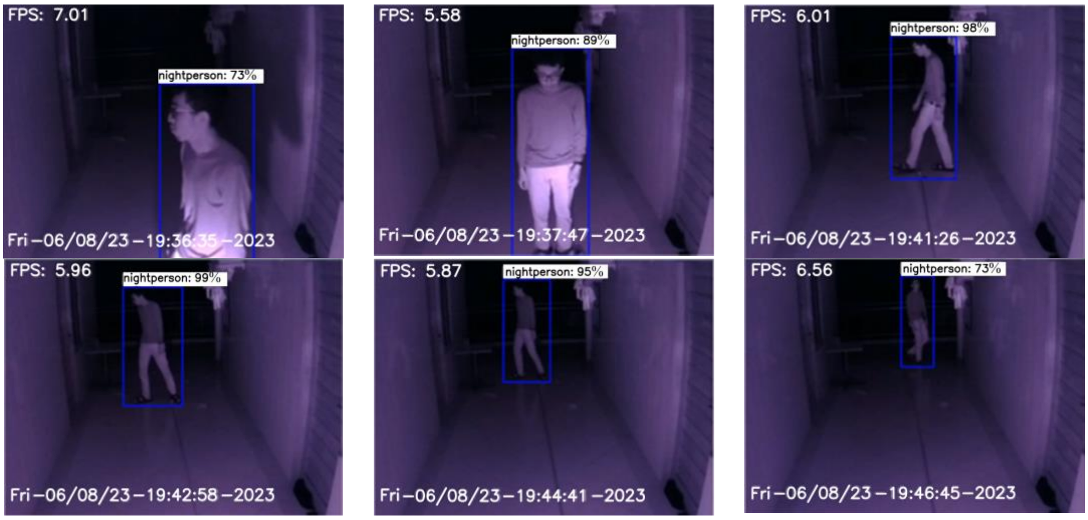

FPS: 7.01
nightperson: 73%
Fri–06/08/23–19:36:35–2023
FPS: 5.58
nightperson: 89%
Fri–06/08/23–19:37:47–2023
FPS: 6.01
nightperson: 98%
Fri–06/08/23–19:41:26–2023
FPS: 5.96
nightperson: 99%
Fri–06/08/23–19:42:58–2023
FPS: 5.87
nightperson: 95%
Fri–06/08/23–19:44:41–2023
FPS: 6.56
nightperson: 73%
Fri–06/08/23–19:46:45–2023

Figure 9. NP with 1 m camera distance from upper-left to 6 m camera distance (bottom-right)

The true NP detection which NP successfully detected with various distances has been summarized in Figure 10. In figure shows the number of true NP detected with various distances in meters. The optimal distance that NP successfully detected falls in the range of 4 m to 6 m. When NP is too close or far from the camera, pixels of NP structure become unclear due to very high or low exposure from an infrared LED source.

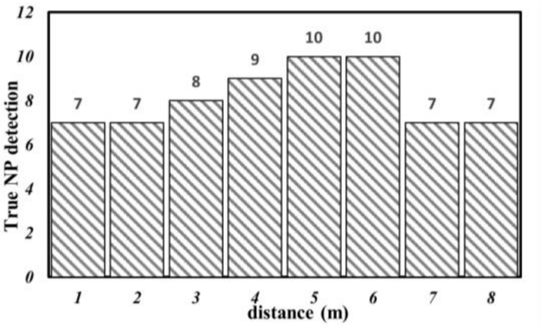


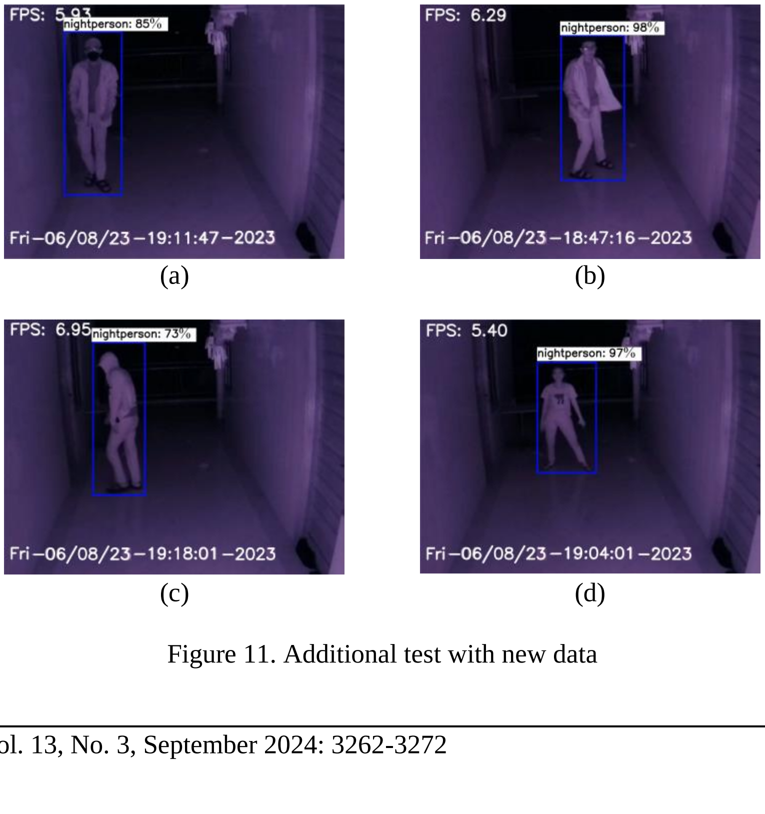
Figure 10. Optimal NP distance from camera successfully detects

The real-time NP detection of precision and recall are computed using (3) and (4), after values of TP, FP, and FN are obtained from the binary confusion matrix. Within 100 times detection of the trial, the result was obtained where TP=86, FP=0, and FN=14. Therefore:

$$Precision = TP/(TP + FP) = 86/(86 + 0) = 1.0$$

$$Recall = TP/(TP + FN) = 86/(86 + 14) = 0.86$$

Both precision and recall are high, which means that the network is capable of generalized detection of NP. An additional test was carried out to confirm the system is working also with absolutely new data. Where NP is in the form of female or male wearing new attributes that have not been in the training dataset. Figure 11 shows the result of NP detection with new data, where Figures 11(a) to 11(c) male wearing a mask, hat, or coat, while Figure 11(d) female. The results of CS show a high above 70% with an average of 6 FPS. This result confirms that the network is capable of generalized detection of NP.


FPS: 5.93
nightperson: 85%
Fri–06/08/23–19:11:47–2023
FPS: 6.29
nightperson: 98%
Fri–06/08/23–18:47:16–2023
FPS: 6.95
nightperson: 73%
Fri–06/08/23–19:18:01–2023
FPS: 5.40
nightperson: 97%
Fri–06/08/23–19:04:01–2023

(a) (b) (c) (d)

Figure 11. Additional test with new data

4. CONCLUSION

The NP images captured by the camera in the dark environment is a day-night or VIS-near infrared camera which operates at 0.4–1.4 μm wavelengths. There is an infrared-cut filter that operates with a mechanical shutter to block infrared light for the day or delivers infrared light for the night or in low-light conditions, providing three-channel high-resolution images. The camera is an active sensor equipped with an infrared light source to illuminate the object's surface and capture back reflection to generate an image. The NP images look dark with predominantly purple, and the pixels of object structure are clear and brighter. The images have a three-channel histogram separated independently and the green channel is dominant. The image histogram assembles to the left toward zero as a dark image. Preparation of the NP detection requires ground truth information within the appropriate format as an additional separated file. The deep learning of object detection uses a pre-train model that is a network consisting of the classifier and the regressor. The classifier has a task to infer classes, while the regressor determines class location with maximum confidence. The average precision is used as a single metric obtained from the precision-recall curve to evaluate the network. The learning curve is evaluated using the train again validation curve. The validation curve over the training curve causes overfitting due to an insufficient set of train datasets. NP detection uses an ordinary camera with infrared capability capable of producing high-precision detection. The additional infrared light source causes objects to be under or overexposed affecting the object being recognized furthermore affecting distance and detection results.

ACKNOWLEDGEMENTS

The author thanks the State Polytechnic of Bali for funding this research. This research contract number is SP.DIPA-023.18.2.677608/2023 Rev.02 (March 27, 2023) from the State Polytechnic of Bali research budget implementation form.

REFERENCES

[1] Z. Wu, B. Hou, B. Ren, Z. Ren, S. Wang, and L. Jiao, “A deep detection network based on interaction of instance segmentation and object detection for sar images,” *Remote Sensing*, vol. 13, no. 13, pp. 1–26, 2021, doi: 10.3390/rs13132582.

[2] Q. Guo, J. Liu, and M. Kaliuzhnyi, “YOLOX-SAR: high-precision object detection system based on visible and infrared sensors for SAR remote sensing,” *IEEE Sensors Journal*, vol. 22, no. 17, pp. 17243–17253, 2022, doi: 10.1109/JSEN.2022.3186889.

[3] L. Tang, W. Tang, X. Qu, Y. Han, W. Wang, and B. Zhao, “A scale-aware pyramid network for multi-scale object detection in SAR images,” *Remote Sensing*, vol. 14, no. 4, pp. 1–24, 2022, doi: 10.3390/rs14040973.

[4] M. Cho, “A study on the obstacle recognition for autonomous driving RC car using LiDAR and thermal infrared camera,” in *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*, 2019, pp. 544–546, doi: 10.1109/ICUFN.2019.8806152.

[5] J. Yin *et al.*, “ProposalContrast: unsupervised pre-training for LiDAR-based 3D object detection,” in *Computer Vision – ECCV 2022*, Cham: Springer, 2022, pp. 17–33, doi: 10.1007/978-3-031-19842-7_2.

[6] G. Zamanakos, L. Tsochatzidis, A. Amanatiadis, and I. Pratikakis, “A comprehensive survey of LIDAR-based 3D object detection methods with deep learning for autonomous driving,” *Computers and Graphics (Pergamon)*, vol. 99, pp. 153–181, 2021, doi: 10.1016/j.cag.2021.07.003.

[7] Z. Feng *et al.*, “Perfecting and extending the near-infrared imaging window,” *Light: Science and Applications*, vol. 10, no. 1, pp. 1–18, 2021, doi: 10.1038/s41377-021-00628-0.

[8] C. Liu *et al.*, “Silicon/2D-material photodetectors: from near-infrared to mid-infrared,” *Light: Science and Applications*, vol. 10, no. 1, pp. 1–21, 2021, doi: 10.1038/s41377-021-00551-4.

[9] K. B. Beć, J. Grabska, and C. W. Huck, “Near-infrared spectroscopy in bio-applications,” *Molecules*, vol. 25, no. 12, pp. 1–36, 2020, doi: 10.3390/molecules25122948.

[10] A. D. Algarni, “Efficient object detection and classification of heat emitting objects from infrared images based on deep learning,” *Multimedia Tools and Applications*, vol. 79, no. 19–20, pp. 13403–13426, 2020, doi: 10.1007/s11042-020-08616-z.

[11] H. Patel and K. P. Upla, “Night vision surveillance: object detection using thermal and visible images,” in *2020 International Conference for Emerging Technology (INCET)*, 2020, pp. 1–6, doi: 10.1109/INCET49848.2020.9154066.

[12] N. Bustos, M. Mashhadi, S. K. L. -Yuen, S. Sarkar, and T. K. Das, “A systematic literature review on object detection using near infrared and thermal images,” *Neurocomputing*, vol. 560, 2023, doi: 10.1016/j.neucom.2023.126804.

[13] Y. He *et al.*, “Infrared machine vision and infrared thermography with deep learning: a review,” *Infrared Physics and Technology*, vol. 116, pp. 1–38, 2021, doi: 10.1016/j.infrared.2021.103754.

[14] C. Jiang *et al.*, “Object detection from UAV thermal infrared images and videos using YOLO models,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 112, pp. 1–17, 2022, doi: 10.1016/j.jag.2022.102912.

[15] P. J. Burke *et al.*, “Overcoming barriers to solar and wind energy adoption in two Asian giants: India and Indonesia,” *Energy Policy*, vol. 132, no. 1, pp. 1216–1228, 2019.

[16] X. Dai, X. Yuan, and X. Wei, “TIRNet: object detection in thermal infrared images for autonomous driving,” *Applied Intelligence*, vol. 51, no. 3, pp. 1244–1261, 2021, doi: 10.1007/s10489-020-01882-2.

[17] I. K. Swardika, P. A. W. Santiary, I. B. I. Purnama, and I. W. Suasnawa, “Development of green zone energy mapping for community-based low carbon emissions,” *International Journal on Advanced Science, Engineering and Information Technology*, vol. 10, no. 6, pp. 2472–2477, 2020, doi: 10.18517/ijaseit.10.6.12642.

[18] Y. Zhang, Y. Zhang, Z. Shi, J. Zhang, and M. Wei, “Design and training of deep CNN-based fast detector in infrared SUAV surveillance system,” *IEEE Access*, vol. 7, pp. 137365–137377, 2019, doi: 10.1109/ACCESS.2019.2941509.

[19] X. Li, S. Qiu, and Y. Song, “Dynamic synopsis and storage algorithm based on infrared surveillance video,” *Infrared Physics and Technology*, vol. 124, 2022, doi: 10.1016/j.infrared.2022.104213.

[20] P. A. W. Santiary, I. K. Swardika, I. B. I. Purnama, I. W. R. Ardana, I. N. K. Wardana, and D. A. I. C. Dewi, “Labeling of an intra-class variation object in deep learning classification,” *IAES International Journal of Artificial Intelligence*, vol. 11, no. 1, pp. 179–188, 2022, doi: 10.11591/ijai.v11.i1.pp179-188.

[21] P. A. W. Santiary, I. K. Swardika, D. A. I. C. Dewi, and I. B. K. Sugirianta, “Intra-class deep learning object detection on embedded computer system,” *IAES International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 430–439, 2024, doi: 10.11591/ijai.v13.i1.pp430-439.

[22] Y. P. Loh and C. S. Chan, “Getting to know low-light images with the exclusively dark dataset,” *Computer Vision and Image Understanding*, vol. 178, pp. 30–42, 2019, doi: 10.1016/j.cviu.2018.10.010.

[23] Y. Xiao, A. Jiang, J. Ye, and M. W. Wang, “Making of night vision: object detection under low-illumination,” *IEEE Access*, vol. 8, pp. 123075–123086, 2020, doi: 10.1109/ACCESS.2020.3007610.

[24] A. Erkan *et al.*, “Influence of headlight level on object detection in urban traffic at night,” *Applied Sciences*, vol. 13, no. 4, pp. 1–21, 2023, doi: 10.3390/app13042668.

[25] M. Schutera, M. Hussein, J. Abhau, R. Mikut, and M. Reischl, “Night-to-day: online image-to-image translation for object detection within autonomous driving by night,” *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 3, pp. 480–489, 2021, doi: 10.1109/TIV.2020.3039456.

[26] J. Yun and J. Woo, “A comparative analysis of deep learning and machine learning on detecting movement directions using PIR sensors,” *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2855–2868, 2020, doi: 10.1109/JIOT.2019.2963326.

[27] M. Verma, R. S. Kaler, and M. Singh, “Sensitivity enhancement of passive infrared (PIR) sensor for motion detection,” *Optik*, vol. 244, 2021, doi: 10.1016/j.ijleo.2021.167503.

[28] G. R. D. Prabhu *et al.*, “Facilitating chemical and biochemical experiments with electronic microcontrollers and single-board computers,” *Nature Protocols*, vol. 15, no. 3, pp. 925–990, 2020, doi: 10.1038/s41596-019-0272-1.

[29] L. Gomes, Z. A. Vale, and J. M. Corchado, “Multi-agent microgrid management system for single-board computers: a case study on peer-to-peer energy trading,” *IEEE Access*, vol. 8, pp. 64169–64183, 2020, doi: 10.1109/ACCESS.2020.2985254.

[30] S. Prongnuch and S. Sitjongsataporn, “Differential drive analysis of spherical magnetic robot using multi-single board computer,” *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 4, pp. 264–275, 2021, doi: 10.22266/ijies2021.0831.24.

[31] S. Du, B. Zhang, P. Zhang, P. Xiang, and H. Xue, “FA-YOLO: an improved YOLO model for infrared occlusion object detection under confusing background,” *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–10, 2021, doi: 10.1155/2021/1896029.

[32] J. Wu, T. Shen, Q. Wang, Z. Tao, K. Zeng, and J. Song, “Local adaptive illumination-driven input-level fusion for infrared and visible object detection,” *Remote Sensing*, vol. 15, no. 3, pp. 1–19, 2023, doi: 10.3390/rs15030660.

[33] M. Kristo, M. I. -Kos, and M. Pobar, “Thermal object detection in difficult weather conditions using YOLO,” *IEEE Access*, vol. 8, pp. 125459–125476, 2020, doi: 10.1109/ACCESS.2020.3007481.

[34] X. Dai *et al.*, “Near infrared nighttime road pedestrians recognition based on convolutional neural network,” *Infrared Physics and Technology*, vol. 97, pp. 25–32, 2019, doi: 10.1016/j.infrared.2018.11.028.

BIOGRAPHIES OF AUTHORS

I Ketut Swardika holding the position of associate professor at the Department of Electrical Engineering in the State Polytechnic of Bali, completed his Doctor of Engineering at Yamaguchi University. He has authored a multitude of educational resources covering a range of subjects including electrical engineering, remote sensing, and computer programming. Additionally, he has authored many journal papers and teaching materials. Currently, his research is centered on capabilities of artificial intelligence for automation and related subjects, also utilizing nighttime remote sensing observations to advance energy and environmental sustainability efforts. He can be contacted at email: swardika@pnb.ac.id.

Putri Alit Widyastuti Santiary is employed as a lecturer within the Department of Electrical Engineering at the State Polytechnic of Bali. With a master’s degree in Computer Engineering from Udayana University, she has devoted more than two decades to instructing computer programming languages at the institution. She has authored multiple educational resources on programming and digital telecommunication in addition to publishing many journal papers. Currently, her research is centered on employing deep-learning techniques for botanical classification. She can be contacted at email: putrialit@pnb.ac.id.