JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

Two Proposed Efficient DET Deep Learning Algorithms based on Vision for Touch less Hand Sanitizer Mobile Robot

Rand Zuhair Khaleel

Informatics Institute for Postgraduate Studies/Iraqi Commission for Computers and Informatics, Iraq.

rand1993zuhair@gmail.com

Firas A. Raheem

Control and Systems Eng. Dept, University of Technology- Iraq, Iraq

firas.a.raheem@uotechnology.edu.iq

Received: 2022 March 15; **Revised:** 2022 April 20; **Accepted:** 2022 May 10

Abstract – The sanitizer dispensers are too useful in a variety of settings to sterilize and protect people from illness, including patients and medical staff in hospitals, teachers, and students in schools or universities, and so on especially in coronavirus. The hand sanitizer dispenser mobile robot is one of the medical applications that is used to detect people who want to sanitize their hands. The object detection-based deep learning technique helps to recognize humans. In this paper, two algorithms based on the EfficientDet deep learning using a camera vision technique for a touchless hand sanitizer mobile robot are proposed. Each algorithm has been 80% trained dataset and 20% validation dataset. The AP is equal to 96% for algorithm-1 and 94% for algorithm-2. In algorithm-2 the first person can sanitize his/her hands and the second person also sanitize his/her hands after the first person. While in algorithm-1 just only one person can sanitize his/her hands. The loss errors of validation and the training of the proposed EfficientDet algorithm-2 are better than the loss errors of the proposed EfficientDet algorithm-2. The proposed EfficientDet algorithm-2 has two classes but the proposed EfficientDet of algorithm-1 has 7 classes. The two proposed algorithms have good AP results. The training time of algorithm-1 = 6 hours while the training time of algorithm-2 = 1 hour. The other proposal step includes changing the input-equal image size to a non-equal size, while the second proposal step presents a new flowchart for a mobile robot hand sanitization application which will be too useful for the COVID-19 pandemic to sanitize each person's hand in public and closed places. Dataset has been created and prepared using image processing and a graphical tool called as LabelImg. The Dataset images were taken using a vision-based Raspberry Pi version 4 mini-computer. Then the dataset has been trained by the EfficientDet algorithm. The Python 3.7.4 programing language has been used. This proposed method has very good results, an efficient and scalable technique for hand sanitization person with a mobile robot application. In future work, the full version of the hardware implantation will be done where the mobile robot will recognize the people using a deep learning approach and move to the person who wants to sanitize her/his hand.

Keywords – Python, Mobile robot, Deep Learning, Raspberry Pi, EfficientDet

1. INTRODUCTION

Artificial intelligence is the most important field that is used to enhance and optimize the different types of robot applications, such as robots called mobile robots controlled by raspberry pi remotely via the internet used for surveillance purposes [1]. A mobile robot with wheels is one robot that moves from one location to another without the assistance of a human operator [2]. In the last three years, the COVID-19 disease has rapidly spread across the world. The virus infection increased and led to people's death. The interaction of people is less in all places [3][4]. Because these reasons, it is necessary needed, to find out hand sanitizer robots/devices to reduce COVID-19 in buildings and general locations such as hospitals, schools, etc. [5]. Deep Learning is one of the good techniques that help robots detect features using object detection. It can detect the objects such as rectangular bounding boxes for illness/normal persons who want for sanitizing their hands [6]. Computer vision with feature extraction has many applications such as recognizing the optical character, object detection for automated checkout lanes, fingerprints, in surgery, object recognition the many diseases, etc. One of the most computer vision techniques used to detect objects in images and videos is called object detection. It interfaced with deep learning in order to produce meaningful and good results [7] [8].

2. RELATED WORK

Many researchers have envisaged the techniques of robot/device sanitizers, especially in coronavirus pandemics. These techniques are classified into two parts. One part studied robot/device sanitizers without deep learning and the other part explained object detection with deep learning.

2.1. A Robot/Device Sanitizers without Deep Learning

The authors in [9] have been designed an automatic hand sanitizing system that compatible with various sanitizer containers. The system used in the market. Several hand sanitizers are automatically pumped. However, because hand sanitizer containers and pump devices are only designed to work with other products from the same manufacturer, consumers must also replace the liquid container if they replace the hand sanitizer. The other researchers [10] emphasized review work for Alcohol-based hand sanitizers (ABHS) to be regarded from a multidimensional perspective from the design level. To ensure ABHS product efficacy and safety, it's critical to understand how these components interact. In another work [11] the use of a mask and frequent hand washing in the event of a coronavirus outbreak has become commonplace in our daily lives, as has social separation. In addition, Temperature is also measured with IR temperature sensors in schools, colleges, malls, and restaurants. The AROGYAKAVACHAM—an automatic hand sanitizer dispenser device with a contactless temperature measurement has been designed.

Furthermore, in [12] the researchers have designed and implemented a low-cost smart hand sanitizer dispenser with a door and controlled by an ATMEGA328P microcontroller. Other authors also, designed an automatic hand wash sanitizer system. The system design has saved cost and power. [13]. Besides that, the hand sanitizer mobile robot is designed to sanitize people's hands [14].

2.2. An Object Detection with Deep Learning

A set of significant researches have studied object detection of deep learning using EfficientDet network-based object detection methods have been studied as the following: in [15] the authors presented a deep learning algorithm called EfficientDet for detection of the cigarette butts of persons who are smoking in public regions and to build a harmonious society in China buildings. It is used the computer vision applications.

An EfficientDet consists of the backbone called Efficientnet network and Bi-Directional Feature Pyramid Network (BiFPN) as a neck network. They are useful for feature extraction and a set of fixed scaling coefficients which is used to scale the network's depth, width, and resolution. The programing language is Python version 3.7.4. The maximum mAP was achieved at 44.32%. In another work [16] the researchers represented the EfficientDet algorithm to detect the wheat ear image. The total dataset of images was 608 taken, and labelImg program was used to label the all images. The training images equal 540 and the testing images equal 68 images. The Python programming language version 3.7. used and the epoch has been set at 200. The average accuracy reached 95.30%. Furthermore, other authors in [17] proposed an algorithm that helps to detect some of the COVID-19 wastes that are inappropriately disposed off. It used its own dataset and compared it with three different architectures of CNN using the transfer learning approach. It has been the best algorithm for the dataset to be as EfficientDet D0 algorithm. The dataset is taken from the Common Objects in Context (COCO). The precision achieved 0.82% mAP.

EFFICIENTDET ALGORITHM

The EfficientDet deep learning method is used in many applications such as computer

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

vision, robot, and object detection. It is defined as the one of deep learning networks proposed by Google [18]. It consists of combing three networks see (Fig. 1) described below.

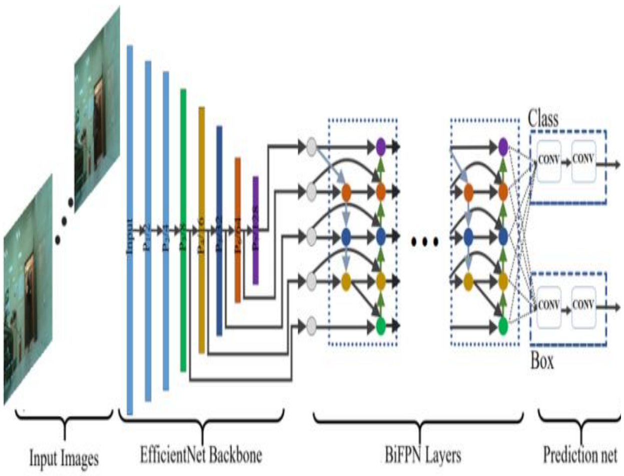

Fig. 1. The proposed EfficientDet algorithm structure.

3.1. The Backbone EfficientNet network

EfficientNet Backbone is the first part of the EfficientDet network that depends on the compound scaling technique. The compound scaling method is defined as uniformly balancing the depth, width, and image resolution of the network and reusing the same depth/width scaling coefficients [18].

3.2. The neck BiFPN network

It is an efficient network that depends on a compound scale feature network, which has been taken from the EfficientNet Backbone network. BiFPN is designed to aggregate features at different resolutions. BiFPN repeatedly applies top-down and bottom-up bidirectional feature fusion. These fused features are fed to a class and box of the network in order to produce the outputs of bounding box and object class predictions respectively. The BiFPN integrates both the bidirectional cross-scale connections and the fast normalized fusion. The input images of backbone network compound scaling of feature levels from 3 to 7 features {P3, P4, P5, P6, P7}. These features are extracted in the next network called BiFPN as seen in Figure 2.2. The input image resolution has been compound scaled by dividing by 27 =128 so that the resolution increased as in

RBiFPN =512+ Ø.128 (1)

Ø is the compound coefficient which controls all scaling dimensions; For example, the two fused features at level 6 for BiFPN can describe as in (2,3). and shown in Fig. 2.

$$P_6^{td} = Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize\,(P_7^{in})}{w_1 + w_2 + \epsilon}\right)$$

(2)

$$P_6^{out} = Conv\left(\frac{w_1' \cdot P_6^{in} + w_2' \cdot P_6^{in} + w_3' \cdot \; Resize\,(P_5^{out})}{w_1' + w_2' + w_3' + \epsilon}\right)$$

(3)

P_6^{td} presented as the intermediate feature at level 6 on the top-down pathway, and P_6^{out} marked as the output feature at level 6 on the bottom-up pathway. P_i^{in} in marked as the feature level with a resolution of $1/2^i$. The all-other features are constructed in the same

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

manner. It can be resized by fusing features of different resolutions to the same resolution and then summing them. The resizing image used the scaling for each dimension by a constant ratio this method is called compound scaling as described in the section above. Therefore, Compound scaling makes sense when the size of the images used to train ConvNets grows greater because larger images require deeper networks to increase the receptive field and more channels to catch tiny features in the larger image. The Pyramid network has a self-attention upsampling in order to recover pixel localization. The feature fusion, besides that addition the batch normalization and activation function after each convolution are useful to increase the efficiency. The compound scaling method of BiFPN has linearly increased the depth using grid search on list values {1.2,1.25,1.3,1.35,1.4,1.45}and pick the best value is 1.35 as the BiFPN so that the width W_{BiFPN} and depth D_{BiFPN} are scaled as equations below [18].

$$W_{BiFPN} = 64 \,.\, (1.35^{\emptyset}) \quad (4)$$

$$D_{BiFPN} = 3 + \emptyset \quad (5)$$

3.3. The Box/class Head predication network

In the last network, the width of predication is always equal to the width of BiFPN ($W_{pred} = W_{BiFPN}$). The depth equation of box and class are equal and it becomes as:

$$D_{box} = D_{class} = 3 + [\emptyset/3] \quad (6)$$

from scaling Equations 1,2,3 with different values of φ from 0 to 7, The EfficientDet-D0 (φ = 0) to D7 (φ = 7) means that the simple scaling method can improve efficiency and accuracy than other single-dimension scaling methods for more details described in [18]. Most studies have developed a method of feature fusion. It is a technique of integrating related information extracted from a group of testing and training the images without losing any image data. The features fusion with various resolutions have been resized to the same resolution and then sum them. All previous studies treated all input features similarly and without distinction. But, the different input features with different resolutions, caused the output unequal feature. This problem was solved by adding additional weight for each input image and letting the network for learning each input feature. Depending on this concept, there are three weighted fusion layers. The first layer named (unbounded fusion) layer can express as:

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

$$O = \sum_i w_i . I_i$$

(7)

$w_i$is learnable weight scalar multiplied with the input I_i. The scalar product makes the learning most accurate with minimum computational costs. But the scalar product weight is also unbounded, i.e it can cause the training to be unstable. So that, each weight must normalize for bounding each weight's value range. The second layer (Softmax based fusion) can be written as a classical softmax equation (7). It is applied to each weight and all weights are normalized between the range 0 to 1. The extra softmax made the GPU system slow therefore the third layer must be implemented with (fast normalization fusion). The fast normalization fusion equation is:

$$O = \sum_i \frac{w_i}{\epsilon + \sum_i w_j} . I_i$$

(8)

$w_i \geq 0$ ensured by Relu function applied after every w_i, and Epsilon (ϵ = 0.001) was used for avoiding numerical instability. Each normalized weight ranged from 0 to 1, but there is no softmax layer operation here so it is more efficient. The fast fusion approach is very fast that runs up to 30% faster on GPU systems [18]. The structure of our algorithm is seen in Fig. 1.

THE AVERAGE PRECISION

The average precision (AP) is a numerical tool used for comparison instead of intersecting curves from precision and recall. AP is a prediction over k number of classes. The mean average precision (mAP) is the mean of AP across all number of classes [19]:

$$mAP = \sum_{i=1}^{k} \frac{AP_i}{k}$$

(9)

THE LOSS FUNCTION

A loss or error function L after training is a mathematical formula used to produce loss values during training time. It measures the difference between predicted values *(ŷ)* and the expected value *(y)* [20].

$$\boldsymbol{L(y, \hat{Y}) = \sum_{i=1}^{n} (y - \hat{Y})}$$

(10)

THE PROPOSED WORK

The proposed work consists of the following:

6.1 The Box/class Head predication network

The dataset has been taken of the people in the university such as academes, students, employers, etc.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

1) creating, preparing, and processing 494 images input images dataset captured from the Raspberry Pi version 2.1 camera for more detail on this camera described in [21] with a mini-computer Raspberry Pi 4 system [22].

2) Then, the graphical image annotations (LabelImg) software has been written in Python language and used to make a bounding box of each object to make many classes. The output of each image file was saved and consisted of two files: one XML file in PASCAL VOC format, and the other, image.jpg. These two outputs file of each image from LabelImg software supports the deep learning algorithms. In this work, the 7 number of classes for object detection were created and classified into following: [left_nearest_person, nearest_human, non_person, person_back, person_front, person_side, right_nearest_person]. These classes are very useful in deep neural network training as in Fig. 2.

Fig. 2. The proposed block diagram of dataset processing.

6.2 The Proposed EfficientDet algorithms

In this work, the proposed EfficientDet is used in the following steps of work:

1) The non-equal size image for example 1024X768 set as input images instead of 512X512 i.e the equal size images. This proposed EfficientDet algorithm can train any size of non-equal size images dataset.

2) In Fig. 1, the EfficientDet structure has been enhanced by achieving Separable Conv class-predict contains 4,671 parameters and Separable Conv class-box contains 2,916 parameters and keras_layer_4 (KerasLayer) equals to 3,234,464 parameters. The total parameters evaluated from sum of trainable = 3,194,915 and non-trainable = 47,136 parameters. The total parameters are equal to 3,242,051.

3) Keras is the high-level API for TensorFlow version 2 that has an approachable, highly-productive interface to solve these deep learning problems [23].

4) Then the training of two algorithms was converted to algorithm lite so that can Raspberry Pi read it and test all the images on the Raspberry Pi system that is attached to the mobile robot for sanitization of the people's hands

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

5) A new flowchart of mobile robot hand sanitation system proposed that is too useful in the COVID-19 pandemic to sanitize the person's hand in public and closed buildings. The proposed flowchart method of this work is explained below as in Fig. 3. It initializes with a selection of the position of the mobile robot. Then images were captured from a mobile robot vision called raspberry pi camera. The ultrasonic sensor distance has been read between the object and the mobile robot. Then these images are saved, processed, and prepared for deep learning as explained in the previous section. After training dataset using proposed EfficientDet deep learning algorithm, if the one person detection or one interested person of two-person detection called [nearest_human] class. Then the outputs are class/box of person position detection and loss computed after training if =< 0.3 the mobile robot moving to the person who wants to sanitize her/his hand. Else if there isn't any person in the place, the non-person class detection, and the mobile robot doesn't move in this case. The training was implemented in the Colab environment by Python 3.7.4 then the algorithm was converted to an extension that can read it from raspberry pi 4.

6) The difference between Figure 3 and Figure 4 is when the dataset has been trained using the proposed EfficientDet algorithm-1 if an interested person number one is trying to pass the door of the laboratory or the room of the main building. In this case, person number one is detected with a class called (nearest_person). While person number two is the person standing behind the door of the room of the building and he is not interested to pass the door. In this case, this person will be not detected because he/she stands far from the door of the room. The output Class/box for person number one is to detect and the mobile robot should move to sanitize his/her hands. In another case, the dataset has been trained using the proposed EfficientDet algorithm-2 where two interested persons standing inside the room.

In this case, the proposed EfficientDet algorithm-2 will detect both of the interested persons with the class (person_1) for the nearest interested person and the class (person_2) for the farthest interested person. The mobile robot then

should move to sanitize the hand of person_1 first, then move to sanitize the hand of person_2 respectively. In the two cases above, the loss (error)=< 0.3. Furthermore, if there is a case of nobody detection, the mobile robot will not move to any place as shown in Figure 4.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

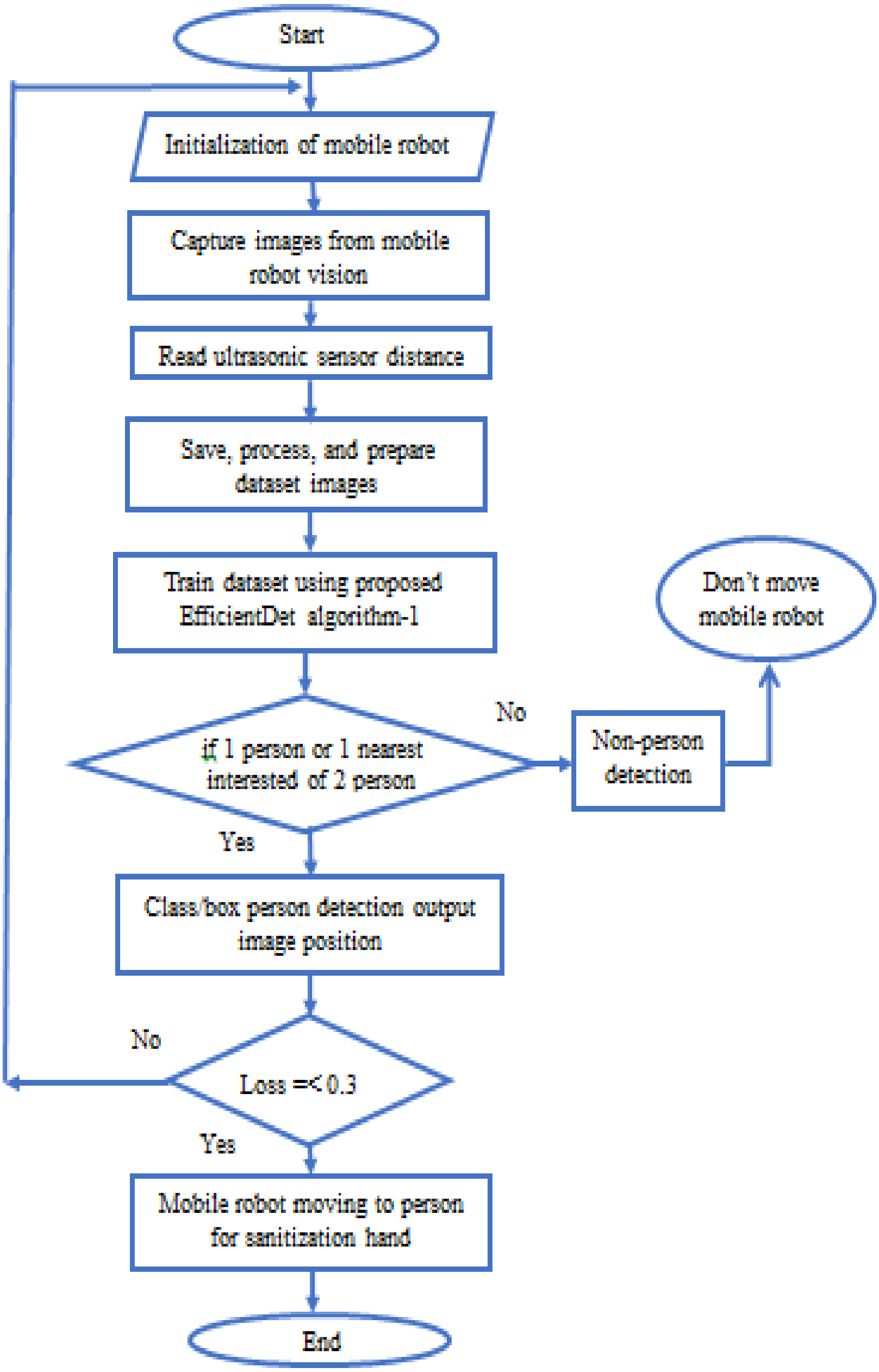

Fig. 3. Flowchart of proposed EfficientDet algorithm-1

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

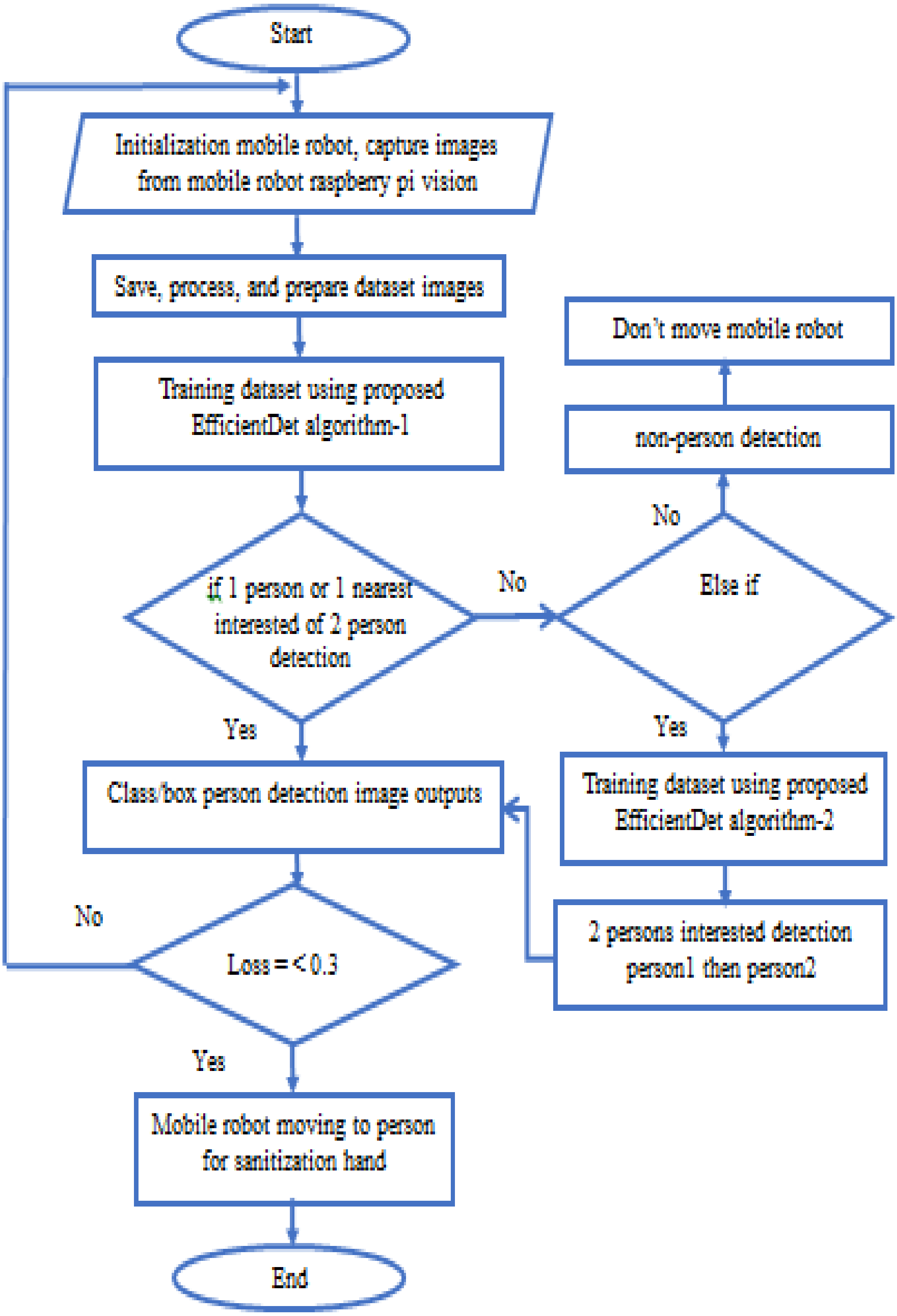

Fig. 4. Flowchart of proposed EfficientDet algorithm-1 and algorithm-2

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

3. Results and Discussion

In this work the results are classified into two parts:

7.1. The Person detection of proposed EfficinetDet deep learning

The testing result after training using the two proposed algorithms with raspberry pi 4 for the case studies has been detected as shown in the Figures below. Fig. 5 shows the AP of the class output right_nearest_person for proposed algorithm-1 which equals 75%.

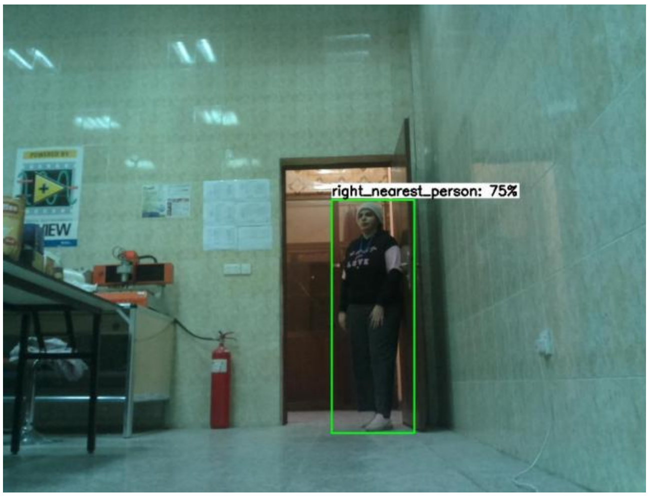

Fig. 5. Case study right_nearest_person detection after training using proposed EfficientDet algorithm-1 with raspberry pi 4 system.

While Fig. 6 shows the AP of the class outputs for two persons detection that achieved 72% for person_1 and 80% for person_2 of proposed algorithm-2.

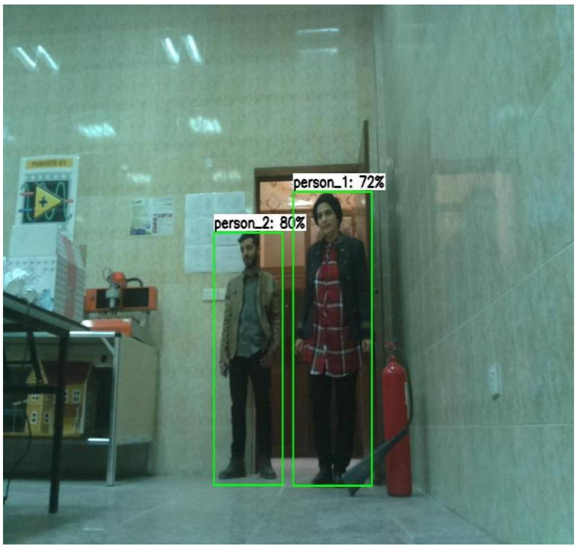

Fig. 6. Case study of person_1 and person_2 detection after training using proposed EfficientDet algorithm-2 of deep learning with raspberry pi 4 system.

7.2. Loss calculations

7.2.1 Loss function calculations of proposed EfficientDet algorithm-1

The Loss is computed according to equation (10). Figure 7 shows the validation VS the training loss results for 100 epochs. The validation loss equals 0.2041 while the training loss is 0.1390. The results show the validation and training errors on 100 epoch are less than the validation and training errors on 50 epoch because if the number of epochs increases the loss decrease.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

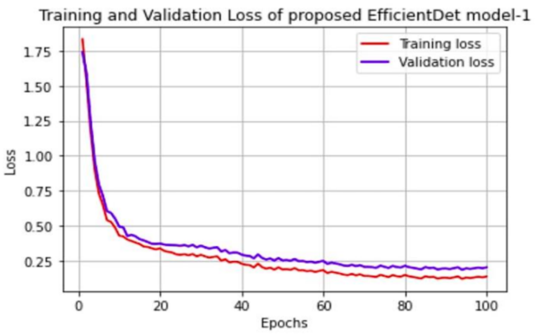

Fig. 7. Training and validation loss results for 100 epochs of EfficientDet algorithm-1.

7.2.2 Loss function calculations of proposed EfficientDet algorithm-2

The training and validation loss for the EfficientDet algorithm-2 deep learning algorithm was calculated for 100 epochs in batch size = 16 due to equation (10). Figure 8 shows the validation VS the training loss results for 100 epochs. The validation loss equals 0.1910 while the training loss is 0.1261 with batch_size=16.

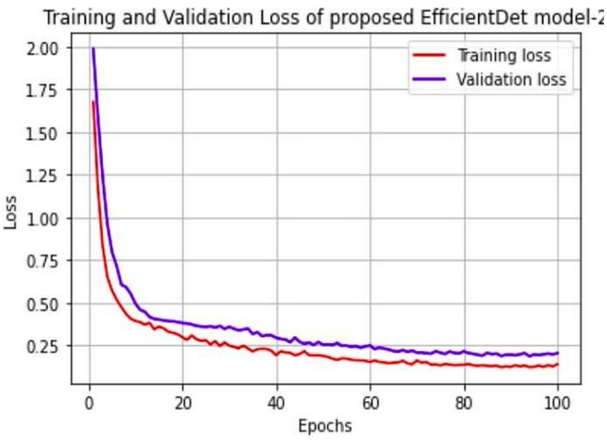

Fig. 8. Training and validation loss results for 100 epochs of EfficientDet algorithm-2.

7.3. Average Precision Calculation

In [24] there is no difference between mAP and AP for example in Common Objects in Context (COCO). COCO is defined as large-scale object detection, captioning dataset, and segmentation. The COCO has 12 features metrics that are used for characterizing the performance of an object detector on COCO. All these metrics are explained in [24].

The precision included the 12 metrics that characterize object detector performance on COCO as mentioned in the previous section calculated due to equation from (9-13)of the EfficientDet deep learning algorithm as in Table 3.1. shows the Average Precision of the EfficientDet deep learning algorithm AP, APIoU=50, APIoU=75) results with different IOU values, AP of 7 algorithm classes (left_nearest_person, nearest_human, non_person, person_back, person_front, person_side, right_nearest_person), AP Across Scale (APlarge), Average Recall (AR) :(ARmax=1, ARmax=10, ARmax=100), and AR Across Scales :(ARlarge). While the AP Across Scales:(APmedium, APsmall), and AR Across Scales :(ARmedium, ARsmall) are equal to (-1) for 100 epochs because there is no detection on small or medium objects.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

Table 1. AP metric character results of algorithm-1 for 100 epochs.

AP metric characters	**AP results of Algorithm-1 (%)**
AP	0.9622481
APIoU=50	0.9802637
APIoU=75	0.9801078
AP (left_nearest_person)	0.9529784
AP (nearest_human)	0.9488293
AP (non_person)	0.937754
AP (person_back)	0.9180553
AP (person_front)	0.92918897
AP (person_side)	0.8974725
AP (right_nearest_person)	0.95145804
APlarge	0.9622507
APmedium	-1.0
APsmall	-1.0
ARmax=1	0.9409995
ARmax=10	0.9537477
ARmax=100	0.9544025
ARlarge	0.9544025
ARmedium	-1.0
ARsmall	-1.0

Table 2. AP metric character results of algorithm-2 for 100 epochs.

AP metric characters	**AP results of Algorithm-1 (%)**
AP	0.9456055
APIoU=50	0.9615361
APIoU=75	0.9644285
AP (person_1)	0.95458984
AP (person_2)	0.93653126
APlarge	0.94365283
APmedium	-1.0
APsmall	-1.0
ARmax=1	0.95480766
ARmax=10	0.9432692
ARmax=100	0.9571154
ARlarge	0.9349396
ARmedium	-1.0
ARsmall	-1.0

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4521-4537
https://publishoa.com
ISSN: 1309-3452

from these results it can be concluded:

1) The loss errors of validation and the training of the proposed EfficientDet algorithm-2 is better than the loss errors of the proposed EfficientDet algorithm-2.
2) The proposed EfficientDet algorithm-2 has two classes but the proposed EfficientDet of algorithm-1 has 7 classes.
3) The detection of two persons in algorithm-2 with high AP as in Table2 AP (person_1) =0.95458984% and AP (person_2)= 0.93653126 while if there are two persons in algorithm-1, the detection in this case only one person detection that called (nearest_human) class. The AP (nearest_human)=0.9488293% as in Table1
4) In algorithm-2 the first person can sanitize her/his hand and the second person also sanitize her/his hand after the first person. While in algorithm-1 just only one person can sanitize her/his hand.
5) Two algorithms have good AP results due to figures 5, 6, and Table1, Table2.
6) The dataset of algorithm-1 is split into two folders called training folder contains (395 images + 395 XML files) and the second folder named validation folder contains (99 images + 99 XML files). The dataset of algorithm-2 is split into two folders called training folder contains (52 images + 52 XML files) and the second folder named validation folder contains (14 images + 14 XML files). Each algorithm has been trained 80% dataset and 20% validation dataset.
7) The training time of algorithm-1 = 6 hours while the training time of algorithm-2 = 1 hour
8) After training, the total parameters of proposed algorithm-1 equal to 3,242,051 while the total parameters of proposed algorithm-2 equal to 3,239,126.

4. Conclusion

COVID-19 spreads around the world. Coronavirus has been classified as a pandemic by the World Health Organization (WHO). It rapidly spread and caused a lot of people to die. Human life has changed, in the way people interact in all areas. This work proposed the EfficintDet method of deep learning by entering the non-equal size of images and it has been proposed flowcharts of person hand sanitization detection with a mobile robot application. The seven classes of AP are evaluated for every class for the proposed EfficientDet algorithm-1 and 2 classes of the proposed EfficientDet algorithm-2. They have excellent AP results as mentioned in the tables above. The detection of two persons in algorithm-2 with high AP as in Table2 AP (person_1) =0.95458984% and AP (person_2)= 0.93653126 while if there are two persons in algorithm-1, the detection in this case only one person detection that called (nearest_human)class. The AP (nearest_human)=0.9488293% as in Table1. In algorithm-2 the first person can sanitize her/his hand and the second person also sanitize her/his hand after the first person. While in algorithm-1 just only one person can sanitize her/his hand. The Python 3.7.4 programing language has been used. The Colab environment is used to train the proposed EfficientDet method of deep learning and then converted to an extension that read from Raspberry Pi 4. After testing, this proposed method has a high enough percentage of AP to make a perfect sense of very good results, an efficient and scalable technique for a person's hand sanitization with a mobile robot application. In future work, we want to implement a mobile robot hand sanitizer that recognizes the people using a deep learning approach and moves to the person who wants to sanitize her/his hand.

5. References

[1] B. K. Oleiwi, "Scouting and controlling for mobile robot based raspberry Pi 3," J. Comput. Theor. Nanosci., vol. 16, no. 1, pp. 79–83, 2019, doi: 10.1166/jctn.2019.7701.

[2] S. G. Tzafestas, Introduction to Mobile Robot Control. 2013.

[3] D. M. Morens, P. Daszak, and H. Markel, "crossm Pandemic COVID-19 Joins History ' s Pandemic Legion," Perspect. Clin. Sci. Epidemiol., vol. 11, no. 3, pp. 1–9, 2020.

[4] L. R. Mbbs, L. B. S. M. D, and M. S. Neurosurgery, "Coronavirus Disease Coronavirus Disease (COVID-19) Spreads," Who, vol. 75, no. 2, pp. 95–97, 2020, [Online]. Available: https://apps.who.int/iris/bitstream/handle/10665/336034/nCoV-weekly-sitrep11Oct20-eng.pdf%0Ahttps://www.who.int/docs/default-source/coronaviruse/situation-reports/20200423-sitrep-94-covid-19.pdf.

[5] Z. H. Khan, A. Siddique, and C. W. Lee, "Robotics utilization for healthcare digitization in global COVID-19 management," Int. J. Environ. Res. Public Health, vol. 17, no. 11, 2020, doi: 10.3390/ijerph17113819.

[6] Youzi Xiao, Zhiqiang Tian, D. X. Lan, Jiachen Yu, Yinshu Zhang, Shuai Liu Shaoyi, and Received:, A review of object detection based on deep learning, vol. 1684, no. 1. Multimedia Tools and Applications, 2020.

[7] R. Szeliski, Computer vision: algorithms and applications. Springer Science & Business Media, 2010.

[8] J. Gao, Y. Yang, P. Lin, and D. S. J. J. o. h. e. Park, "Computer vision in healthcare applications," vol. 2018, ed: Hindawi, 2018.

[9] J. Lee, J. Y. Lee, S. M. Cho, K. C. Yoon, Y. J. Kim, and K. G. Kim, “Design of automatic hand sanitizer system compatible with various containers,” Healthc. Inform. Res., vol. 26, no. 3, pp. 243–247, 2020, doi: 10.4258/hir.2020.26.3.243.

[10] K. Abuga and N. Nyamweya, “Alcohol-Based Hand Sanitizers in COVID-19 Prevention: A Multidimensional Perspective,” Pharmacy, vol. 9, no. 1, p. 64, 2021, doi: 10.3390/pharmacy9010064.

[11] Sathwara, H., Vaghela, P., & Joshi, S. (2022). AROGYAKAVACHAM—Automatic Hand Sanitizer Dispenser with Temperature Measurement. In Proceedings of the International e-Conference on Intelligent Systems and Signal Processing (pp. 785-796). Springer, Singapore.

[12] E. Edozie, W. Janat, and Z. Kalyankolo, “Design and Implementation of a Smart Hand Sanitizer Dispenser with Door Controller using ATMEGA328P,” Int. J. Eng. Inf. Syst., vol. 4, no. 6, pp. 14–18, 2020.

[13] M. M. Srihari, “Self-Activating Sanitizer with Battery Imposed System for Cleansing Hands,” Proc. 2nd Int. Conf. Inven. Res.Comput. Appl. ICIRCA 2020, pp. 1102–1105, 2020, doi: 10.1109/ICIRCA48905.2020.9183347.

[14] Hind. Z. Khaleel, Firas. A. Raheem, and M. Azzam, “Design and Implementation of a Touchless Hand Sanitizer Dispenser Mobile Robot,” vol. 01012, pp. 1–8, 2022.

[15] Liao, Jianhao, Jiayu Zou, Ao Shen, Jinfu Liu, and Xiaofei Du. "Cigarette end detection based on EfficientDet." In Journal of Physics: Conference Series, vol. 1748, no. 6, p. 062015. IOP Publishing, 2021.

[16] Cao, Liangben, Xixin Zhang, Jingyu Pu, Siyuan Xu, Xinlin Cai, and Zhiyong Li. "The Field Wheat Count Based on the Efficientdet Algorithm." In 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE), pp. 557-561. IEEE, 2020.

[17] Buragohain, Ayushman, Bhabesh Mali, Santanu Saha, and Pranav Kumar Singh. "A deep transfer learning based approach to detect COVID-19 waste." Internet Technology Letters 5, no. 3 (2022): e327.

[18] Tan, Mingxing, Ruoming Pang, and Quoc V. Le. "Efficientdet: Scalable and efficient object detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10781-10790. 2020.

[19] Cowton, Jake, Ilias Kyriazakis, and Jaume Bacardit. "Automated individual pig localisation, tracking and behaviour metric extraction using deep learning." IEEE Access 7 (2019): 108049-108060.

[20] D. O. Melinte, A.-M. Travediu, and D. N. J. A. S. Dumitriu, "Deep convolutional neural networks object detector for real-time waste identification," vol. 10, no. 20, p. 7301, 2020.

[21] https://www.raspberrypi.com/. last accessed 2022/4/27.

[22] https://www.tradeinn.com/techinn/en/raspberry-pi-camera-v2/137489672/p. last accessed 2022/4/23.

[23] https://www.tensorflow.org/resources/learn-ml?gclid=Cj0KCQjw-daUBhCIARIsALbkjSZpnk7mHV-nBqTSGFmQdNJZdon1uUWHrD1aUamBgu1rznBZVwPKuoIaAnwhEALw_wcB. last accessed 2022/5/30.

[24] https://cocodataset.org/#detection-eval. last accessed 2022/4/27.