VAST

Vietnam Academy of Science and Technology

Vietnam Journal of Marine Science and Technology

journal homepage: vjs.ac.vn/index.php/jmst

Deep based learning: ebin sorting system development

Dang Vu Kim Ky[1,2], Nguyen Huynh Thong[2,3,*], Au Thuy An[2,3], Pham Tan Hung[2,4]

[1] *Faculty of Electrical and Electronics Engineering, Ho Chi Minh city University of Technology, Ho Chi Minh city, Vietnam*
[2] *Vietnam National University Ho Chi Minh city, Ho Chi Minh city, Vietnam*
[3] *Faculty of Geological and Petroleum Engineering, Ho Chi Minh city University of Technology, Ho Chi Minh city, Vietnam*
[4] *Faculty of Environment and Natural Resources, Ho Chi Minh city University of Technology, Ho Chi Minh city, Vietnam*
[*] E-mail: nhthong@hcmut.edu.vn

Received: 1 January 2022; Accepted: 15 March 2022

ABSTRACT

Vietnam was the fifth country that dumped more plastic into the oceans than the rest of the world combined. The country produces an average of 25.5 million tons of waste per year, of which 75% is buried. Several burial sites in major cities such as Hanoi, Ho Chi Minh city, and Da Nang city are overloaded and negatively affecting citizens' lives. Therefore, sorting at source would significantly reduce soil resources and groundwater pollution and save money on collection costs, transportation, treatment, and easing the pressure on the landfills. Using computer vision may help classify easily recyclable trash into different categories based on its material. This paper proposes an experiment on a customed Convolution Neural Network (CNN) based waste detector for three classes of trash: Plastic bottles, Cans, and Glass. The study used up to 900 hand-collected image data separated into 500 for analysis and 400 remains for assessing the result. All trashed objects used for generating data were collected around the Ho Chi Minh city University of Technology (HCMUT, Vietnam). The Deep Learning model used in this research, named SSD MobileNet V2, is part of the open-source Tensorflow Object Detection API. The system gave the result, the relative match percentage of different amounts of data we fed the model. The system showed linear relativity between the amount of data trained AI100, AI200, AI300, AI400, AI500, and the mean Average Precision (mAP) when testing the system. From this on, the study conducted another experiment to ensure that the performance of the trained model would meet expectations, AI401. In this experiment, the system showed over 80% inaccuracy overall. By back-analysis, this paper would be a good vision for researchers to study and enhance the system's accuracy to serve the purpose of trash classification in innovative trash bin applications.

Keywords: Convolution neural network, deep learning, trash classification, computer vision, tensorflow object detection API.

Citation: Dang Vu Kim Ky, Nguyen Huynh Thong, Au Thuy An, and Pham Tan Hung, 2022. Deep based learning: ebin sorting system development. *Vietnam Journal of Marine Science and Technology*, *22*(1), 85–93. https://doi.org/10.15625/1859-3097/16928

INTRODUCTION

At present, citizens' awareness about sorting trash is a terrible problem, although the generated rate of waste is getting higher. According to the Ministry of Environment and Natural Resources, Ho Chi Minh city produced about 5,500 tons/day of municipal solid waste in 2004 (VMENR, 2004). In 2014, the amount of waste had increased up to 7,000 tons/day with an average generated rate of 1.02 kg/capita/day [1]. The municipal solid waste in Ho Chi Minh city consists of about 80% organic waste; the remaining is an inorganic waste. Approximately 85% of total waste mostly stayed in Phuoc Hiep and Da Phuoc landfill sites; only 15% of waste was recycled. The primary type of waste recycling is the rate of paper at 112 tons/day, plastic at 126 tons/day, and glass at 13 tons/day, respectively [2]. Moreover, the waste treatment system is obsolete. Waste sorting is still manual, mainly dumped in landfills, or some households burn the trash directly.

In modern life, Artificial Intelligence (AI) has played a significant role in many fields to help humans solve minor to enormous global problems. For instance, the application of AI exists in every aspect of life, such as medical, health, agriculture, object detection, etc. Attempts to use Deep Learning in trash classification have become extremely common in the last decade because the development of technology and waste has been an urgent global issue [3–6]. This paper mainly focuses on training a custom trash classification model and assessing its accuracy as a reference when it comes to using pre-trained Deep Learning models in the future.

So far, the applications of AI in sorting trash have become more popular and practical globally, contributing to minimizing the amount of waste generated. However, in Vietnam, especially in Ho Chi Minh city, the applications of AI in trash classification are still limited, and the published dataset is not appropriate for the Ho Chi Minh city situation because there are organic and inorganic trash bins or glass, metal, and can trash bin.

Therefore, this study proposes a customed Deep Learning waste classification based on the CNN algorithm. The model is trained with the dataset of photos of three classes of waste collected on the Ho Chi Minh city University of Technology (HCMUT, Vietnam) campus. This classification model detects different types of garbage and assesses the optimized point in the quantity of data used to train to achieve the desired accuracy, serving as a foundation gradually contributing to the field of research into pre-trained Deep Learning model for specific purposes instead of preparing a model from scratch; ultimately, devoted to solving the problem of sorting waste at source.

DATA MATERIAL

Existing Data

Tensorflow object detection API

Artificial Intelligence used in this paper is a form of API (Application Programming Interface), or it can be understood as an open-source package provided for developers to operate, improve, or integrate into a larger project. Accordingly, Tensorflow - part of Google Inc., offers several AI Deep Learning and Machine Learning models for developers in the form of an open-source API.

As a result, in this paper, we used Object Detection API, which, as the name suggested, is an open-source package used to identify objects instead of building from scratch. We chose a model with stable accuracy and fast processing time because the goal was to apply and integrate it into the device to perform object recognition in real-time. There are some criteria to select the appropriate AI model for different purposes, such as being biased toward the accuracy AI model, AI for image processing, AI for speech recognition, AI text processing, etc. The AI model we chose in this paper is SSD MobileNets V2. This model has the data processing workflow described as the following segment.

MobileNets V2 architecture

The research used SSD MobileNets V2 (Figure 1) that Sandler (2019), was an enormous leap in performance from its previous counterpart. By including Linear Bottleneck and Inverted Residuals [7] in connections between thin convolution layers, the former process the model's intermediate inputs and outputs while the inner layers encapsulate the model's ability to transform from lower-level concepts such as pixels to a higher level of description such as image category or picture classification.

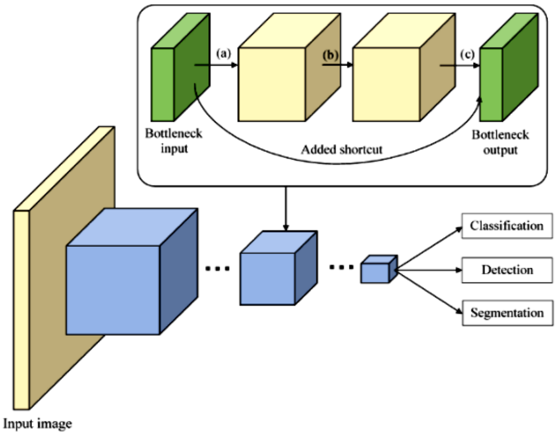


Figure 1. MobileNets V2 built upon the architecture of MobileNets [8] composing linear bottleneck between layers and shortcut connections between bottlenecks. The whole mentioned previously represents a composite convolutional building block of the entire MobileNets V2 architecture. Note: (a) is 1×1 convolution, ReLU; (b) is depthwise convolution of 3×3, ReLU; (c) is 1×1 convolution, linear combination

Evaluation

In object detection, AP (Average Precision) is a factor standard for evaluation in the field of object detection competitors [9–11] like Faster R-CNN, YOLO, Mask R-CNN, etc. Precision measures how accurate object predictions are, which means the percentage of object predictions is correct, following IoU is equated ratio of Overlap area and Union area.

Primary Data

Dataset preparation

A waste area with many small-to-medium-sized canteens operates all day, conducting fairly enough samples for teachers and students of HCMUT to collect data. After surveying all the trash bins in the area, the authors summarized the three main types of trash, namely Cans, Plastic Bottles, and Glass which fit the aim for recyclable waste [12].

This paper hand-collected a dataset containing 3 classes: a plastic bottle, can, and glass, followed the data augmentation techniques when taking the images to provide variation in the dataset (Figure 2). The method included random rotation, lighting, brightness, and scaling of photos to maximize the dataset, ach class has approximately 15 objects shot at different angles and trajectories. There are 500 images in total for the training dataset in Table 1. Objects used in this paper are those common beverages such as pure water bottles, fruit juice bottles made of both glass and plastic, and carbonate water cans. Each object was placed on white cardboard as a background and took the picture in natural room light conditions. The photos were taken by several mobile phone devices, such as Samsung Galaxy Note 8, iPhone X, etc., then were resized into a "standard" of 640×640 pixels which suits the required input data of the used model.

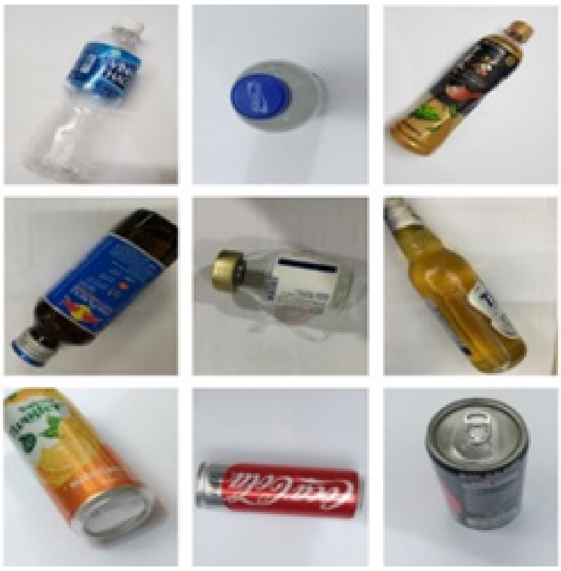


Figure 2. An example of a training dataset with objects representing Plastic bottle, Can, and Glass classes

Table 1. Quantity of images in training dataset

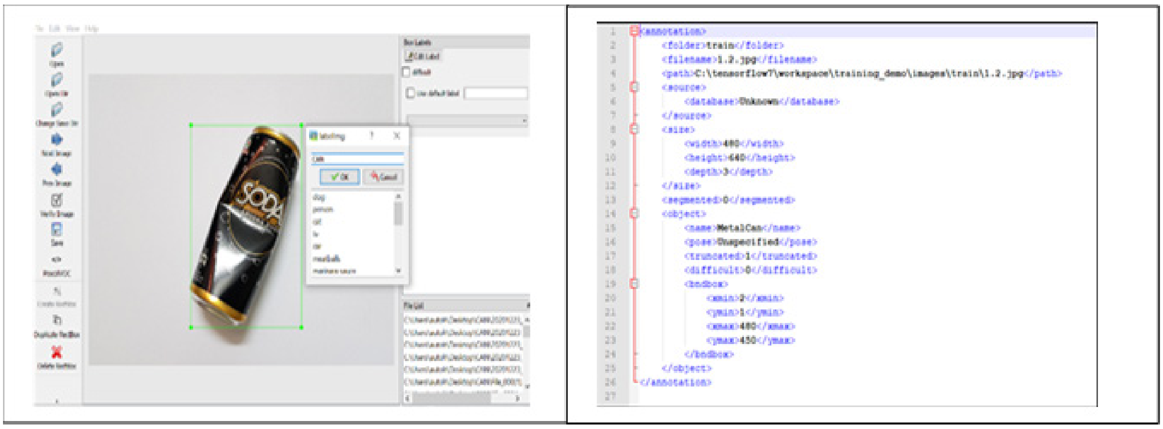
No.	Classes	Number of Objects	Number of Images	Resized image input
1	Plastic Bottle	14	166	640×640 pixels
2	Can	15	166	
3	Glass	17	168	
Total		48	500	

Dataset separation

The authors separated the 500 data images into smaller sets; those sets contain photos from the original 500 but in smaller quantities. The requirement is to keep the same total number of objects in each class. Thus, for set 100 images, authors randomly chose two photographs for one entity; the same goes for sets 200, 300, and 400 with the increasing photos for each object. Thus, the model will be trained with the same number of entities, keeping at 1:1:1, to ensure the trained model is unbiased.

Dataset labeling

The object in an image needs to be rounded like a box and named with a specific label defined. The photos need to be labeled before going into the training process, and a tool called LabelImg [13] is a graphical image annotation tool is used (Figure 3). It is written in Python and uses Qt for its graphical interface. The annotations are saved as XML files in PASCAL VOC format. In the end, for each object, there will be an XML file containing the label of the object and coordinates of the Grounding box that bound the object (Figure 3).



```
<annotation>
    <folder>train</folder>
    <filename>1.2.jpg</filename>
    <path>C:\tensorflow7\workspace\training_demo\images\train\1.2.jpg</path>
    <source>
        <database>Unknown</database>
    </source>
    <size>
        <width>480</width>
        <height>640</height>
        <depth>3</depth>
    </size>
    <segmented>0</segmented>
    <object>
        <name>MetalCan</name>
        <pose>Unspecified</pose>
        <truncated>1</truncated>
        <difficult>0</difficult>
        <bndbox>
            <xmin>2</xmin>
            <ymin>1</ymin>
            <xmax>480</xmax>
            <ymax>450</ymax>
        </bndbox>
    </object>
</annotation>
```

Figure 3. Object (can class) labeled using "LabelImg" (left); XML file with the label name and coordinate of Grounding box (right)

EXPERIMENT AND RESULT

Experiment

This research tends to analyze and evaluate the detection ability of the trained model, using different data sets but with the same hyperparameters as shown in Tables 2, 3. The hyperparameters were set for the pre-trained model conducted on the COCO dataset. The research group kept all these values because they were already suited and applied to those models to analyze and evaluate (Figure 4).

Table 2. Input parameters of training

Parameters	Values
Initial learning rate	0.08
Momentum	0.9
Decay	0.997
Mini batch size	6
Number of epochs	833
Maximum steps	50000
Weight	4e - 05
Matched IoU threshold	0.6

Table 3. Training models with marginal condition

Model deep learning (SSD MobileNets V2)	Training data set	Marginal condition	Output (called name)
Model with set hyperparameters	100 Images	Normalization Loss ≈ 0.1 ± 3%	AI100
	200 Images		AI200
	300 Images		AI300
	400 Images		AI400
	500 Images		AI500

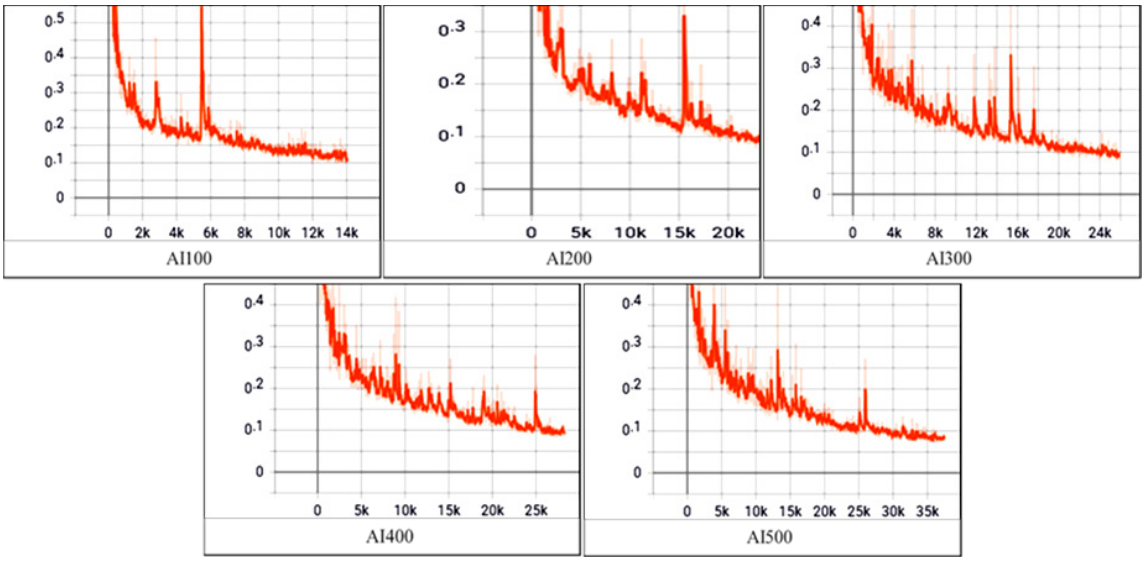


Figure 4. Normalization loss during training and tuning process of loss function value of the network for each model with the naming represents the data set that each was trained with: AI100 (as the model was trained with set 100 images), AI200, AI300, AI400, AI500 respectively

Output result

Dataset training result

After training, the model was ultimately validated using Tensorflow's Object Detection API with provided package for final evaluation. The research group evaluated two different test sets: one containing 100 images, the other containing 150 images, and both had various photos of the same object as in each training set. The research group can then calculate 2 individual Linear Regression lines about the relationship between mAP and the images trained. Besides, there were also statistics about the time needed for training and the incremental steps required for activity. The same for mAP, the research group also analyzed the relationship between the time used for training and the number of images, phases of the training process, and pictures. Table 4 shows the numbers and percentages of the sample in a validation set with different threshold values.

Table 4. Models evaluating on 2 test sets with time and steps needed for training

Images (x)	mAP in set 150 images (y_1)	mAP in set 100 images (y_2)	Time (min.) (t)	Total steps (z)
AI100	50.2%	58.3%	2 hours	14,000
AI200	65.1%	65.6%	3 hours 22 minutes	21,000
AI300	75.2%	75.5%	3 hours 51 minutes	25,880
AI400	81.4%	81.0%	4 hours 36 minutes	28,300
AI500	81.6%	84.1%	5 hours 31 minutes	37,400

Result of finding optimization value

With values produced, the research group outlined the graphs as in Figure 5, with the horizontal axis (x) representing the number of images used for training and the vertical axis (y) representing for variables such as mAP (%), time (minutes), step (steps). With the data points conducted from the experiment, the group determines their Linear Regression.

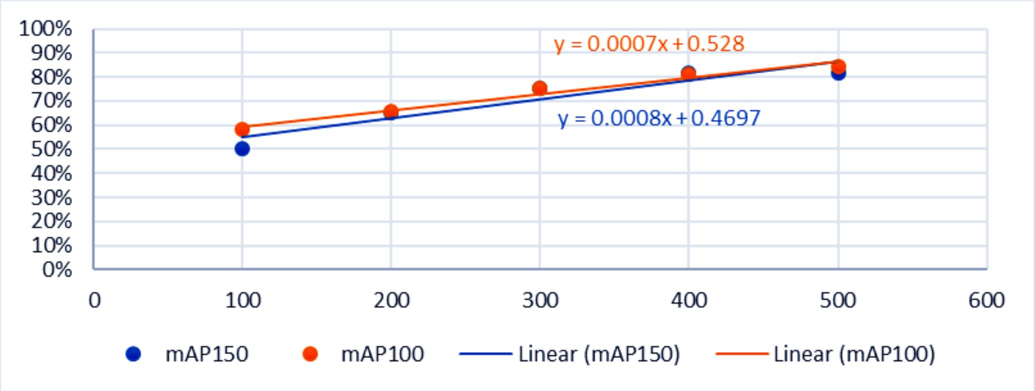


Figure 5. Correlative diagram between mAP (%, (y)) and images used for training (images, (x)). There are 2 lines due to their independence, with the orange line representing for evaluation on the set of 100 images, the blue line represents for evaluation on the set of 150 images

In these 5 pairs of correlative values (y_1, y_2), the paper mainly focused on the correlation between mAP and the Total number of images used for training. The group concluded that the difference between the 2 results Linear Regression is not mentionable when using trained models for evaluation on different test sets. Followingly, calculation on optimization with the requirement of reaching mAP 80% on accuracy to receive the images required for training, Time, and steps needed to train the model. We have:

For the test set of 100 images: $y = 0.0007x + 0.528$ and $R_2 = 0.9669 \Rightarrow x_1 \sim 389$ images.

For the test set of 150 images: $y = 0.0008x + 0.4697$ and $R_2 = 0.8873 \Rightarrow x_2 \sim 413$ images.

Then from x_1 and x_2, we can take the average value of both to get $\overline{x}$ due to these 2 Linear Regression are asymptotic to each other; thus, the error is neglectable, with $\overline{x} = 401$ images. Then from x_1 and x_2, we can take the average value of both to get $\overline{x}$ due to these 2 Linear Regression being asymptotic; thus, the error is neglectable, with $\overline{x} = 401$ images. Similarly, we can receive analytically optimized for t: $t \sim 282$ minutes and k: $k \sim 30{,}780$ steps for some steps. This process would be the foundation for the research group to justify the optimized training ability of the model, from that conduct experiment to verify the trust level of accuracy for entirely new images of the same set of objects with the number of images for each group at 100, 200, 300, 400 to remeasure all the values of mAP, time and steps.

Assessing the capability of detecting objects

In this section, after training a new model with the value of images calculated above, the research group exported a model with some other difference between calculated data and actual data of time, step (Table 5). We can see the remarkable difference in the value of time used for training and a very minimal difference in the steps needed for training between calculated and actual values (Table 6).

The experiment proves that despite the same number of images used for training, the process between consecutive sequences is different; not always training needs the same amount of time and steps, despite the same input and marginal condition.

After exporting the model trained with 401 images, the researchers assessed the new model’s accuracy. They collected the test set for the same number of objects different from those used for training and evaluated the abovementioned model. These variations can be an angle where the objects were photographed, their position inside the picture, the distance between the lens and object, etc. (Table 5).

Table 5. Training models with marginal condition and calculated number of images used for training

Model deep learning (SSD MobileNets V2)	Training data set	Marginal condition	Output	
Model with set hyperparameters	401 Images	Normalization loss ≈ 0.1 ± 3%	AI401	Steps = 34,300
				Time = 445 minutes

Table 6. Different between predicted value and actual value when we reconducted the experiment

	Predicted value	Actual value	Difference (%)
Time	282 minutes	445 minutes	57.8%
Steps	30,780 steps	34,300 steps	11.4%

The difference between these sets is to the human eye, it is not much, but through computer vision and how machines interpret images, these images are entirely separable although there are taken from the same object. Thus, this experiment can evaluate a completely new data set; we can assess the model's performance in Tables 6, 7.

The same with the section above, comparison between the predicted value and actual value, the paper expected the model to reach the wanted value at 80% of mAP. The authors can separate all the measured values into 2 areas: desired area_empty color and undesired area_pink color (Figure 6).

Table 7. mAP measured (in %) by testing the theoretically optimized model on a different test set

Test set image	Set 60	Set 100	Set 200	Set 300	Set 401
mAP	90.2%	85.3%	86.2%	84.3%	81.1%

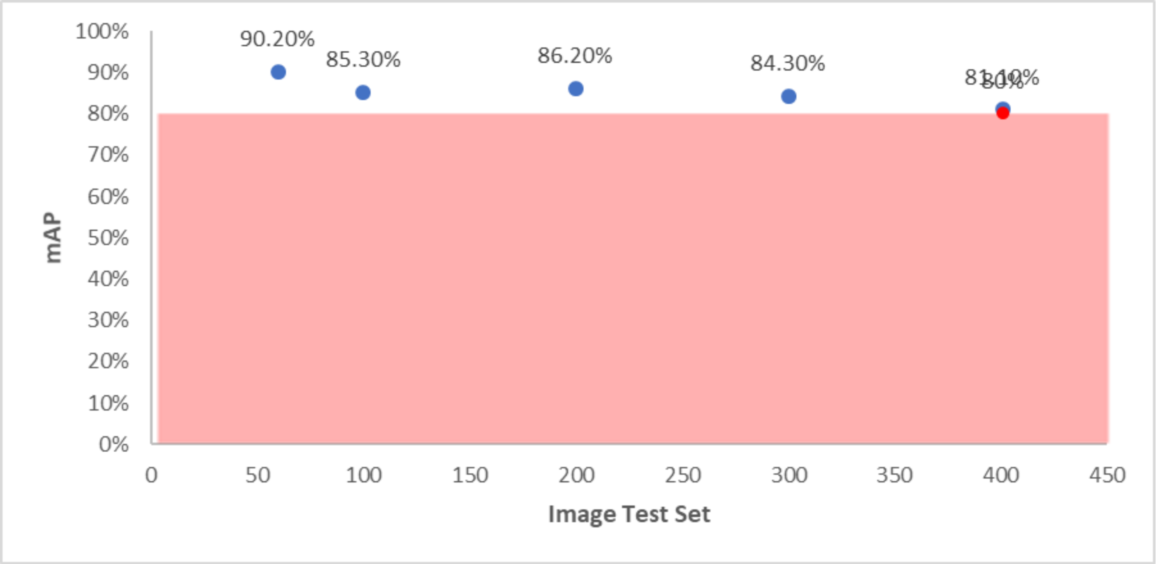


Figure 6. Measurements of mAP on different test sets with the plotted desired and undesired area; the latter plotted in pink below the desired area (dotted)

CONCLUSION

This paper proposes an experiment on training a custom object detector by using Tensorflow Object Detection API. Firstly, the authors have contributed a dataset of 48 objects with up to 900 images collected in the vicinity of HCMUT - Vietnam. Secondly, the research group has trained the model SSD MobileNets V2 with the specified condition and tested it on different sets to show the linearity of the accuracy to the amount of data and the independence of the accuracy to the size of the test set. Finally, with AI401, all test set images always get a minimum of 80% mAP for three classes - plastic bottle, can, glass with 48 objects, and authors experimented with assessing the accuracy of the given model.

This paper also emphasizes that the first attempt of doing this research contributes to solving an urgent environmental problem - waste management. In future work, the study will continuously develop the results to improve the accuracy and prove the feasibility of applying it in small portable devices such as smart trash bins.

DISCUSSION

In recent years, the amount of solid waste in urban areas has gone beyond the processing capacity of the waste treatment system and hasn’t been able to implement modern technologies such as AI, IoT into the current waste management process. Thus tackling these predicaments will ultimately contribute to the work of environment conservation and using the expenses and effort more efficiently by applying the Deep Learning model to the waste management process.

In this study, the authors measured the performance of the trained Deep Learning model with different data materials to construct the theoretical reference and back-analyzed with the reference point to ensure the performance. By applying the framework of Tensorflow Object Detection API, the research group approached the matter in a less sophisticated way, which narrows its distance from theoretical science and practical applications. Ultimately, the result in this research provides a reference for researchers in collecting data material for training models without massively considering external factors; researchers are only required to provide several data images of numbers of objects to achieve a desirable result outcome.

The majority of studies in AI in Vietnamese region show limitations on directly impacting the real-life matters of environment conservation and people’s consciousness. This study also comes short of the idea that the Glass class can be more profound by dividing different glass-material objects with different characteristics: glass from the bottles, bow, mirror, etc. can be separated from dedicated glass such as tempered glass, UV glass, etc. [14, 15]. That can greatly contribute to the glass treatment process in industrial factories.

Acknowledgments: This research is done by the BKI Innovation competition in HCMUT. We acknowledge the support of time and facilities from HCMUT, VNU-HCM for this study.

REFERENCES

[1] DONRE (Department of Natural Resource and Management)-Ho Chi Minh City, 2014. The report on overview of the SWM system of HCMC, Vietnam.

[2] Verma, R. L., Borongan, G., and Memon, M., 2016. Municipal solid waste management in Ho Chi Minh City, Viet Nam, current practices and future recommendation. *Procedia Environmental Sciences*, *35*, 127–139. doi: 10.1016/j.proenv.2016.07.059

[3] Panwar, H., Gupta, P. K., Siddiqui, M. K., Morales-Menendez, R., Bhardwaj, P., Sharma, S., and Sarker, I. H., 2020. AquaVision: Automating the detection of waste in water bodies using deep transfer learning. *Case Studies in Chemical and Environmental Engineering*, *2*, 100026. doi: 10.1016/j.cscee.2020.100026

[4] Bai, J., Lian, S., Liu, Z., Wang, K., and Liu, D., 2018. Deep learning based robot for automatically picking up garbage on the grass. *IEEE Transactions on Consumer Electronics*, *64*(3), 382–389. doi: 10.1109/TCE.2018.2859629

[5] Costa, B. S., Bernardes, A. C., Pereira, J. V., Zampa, V. H., Pereira, V. A., Matos, G. F., Soares, E. A., Soares, C. L., and Silva, A. F., 2018. Artificial intelligence in automated sorting in trash recycling. In *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional* (pp. 198–205). *SBC*. https://doi.org/10.5753/eniac.2018.4416

[6] Vo, A. H., Vo, M. T., and Le, T., 2019. A novel framework for trash classification using deep transfer learning. *IEEE Access*, *7*, 178631–178639. doi: 10.1109/ACCESS.2019.2959033

[7] Yu, J., Jiang, Y., Wang, Z., Cao, Z., and Huang, T., 2016. Unitbox: An advanced object detection network. In *Proceedings of the 24th ACM international conference on Multimedia* (pp. 516–520). https://doi.org/10.1145/2964284.2967274

[8] VMENR (Vietnam Ministry of Environment and Natural Resources), 2014. Canadian International Development Agency, Vietnam Environment Monitor: Solid Waste, Vietnam.

[9] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A., 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision*, *88*(2), 303–338. doi: 10.1007/s11263-009-0275-4

[10] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L., 2014. Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740–755). *Springer, Cham*. https://doi.org/10.1007/978-3-319-10602-1_48

[11] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L., 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252. doi: 10.1007/s11263-015-0816-y

[12] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., and Zheng, X., 2016. TensorFlow: A system for large-scale machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)* (pp. 265–283).

[13] Shi, L., Zhong, H., Xie, T., and Li, M., 2011. An empirical study on evolution of API documentation. In *International Conference on Fundamental Approaches To Software Engineering* (pp. 416–431). *Springer, Berlin, Heidelberg*. doi: 10.1007/978-3-642-19811-3_29

[14] Blengini, G. A., Busto, M., Fantoni, M., and Fino, D., 2012. Eco-efficient waste glass recycling: Integrated waste management and green product development through LCA. *Waste management*, *32*(5), 1000–1008. doi: 10.1016/j.wasman.2011.10.018

[15] Vellini, M., and Savioli, M., 2009. Energy and environmental analysis of glass container production and recycling. *Energy*, *34*(12), 2137–2143. doi: 10.1016/j.energy.2008.09.017