IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

SOLID WASTE SEGREGATION USING YOLO

Ayush Kumar
Department of Computing Technologies,
SRM Institute of Science and Technology,
Kattankulathur, Tamil Nadu, India a

Mihr Anand
Department of Computing Technologies,
SRM Institute of Science and Technology,
Kattankulathur, Tamil Nadu, India

Dr. B. Baranidharan
Department of Computing Technologies,
SRM Institute of Science and Technology,
Kattankulathur, Tamil Nadu, India

Abstract

In an average Indian capital city, the daily production of unsegregated waste exceeds 5000 tons. Despite a doubling of this garbage production rate in the last 5 years, the methods employed for waste management still rely on manual labor. This project proposes a more advanced approach by incorporating smart technology, specifically leveraging machine learning strategies such as Convolutional Neural Network (CNN) to analyze waste content. The model utilizes the You Only Look Once (YOLO) image processing framework to categorize garbage into recyclable and non-recyclable waste. The information generated is then transmitted to a physical robotic arm, responsible for depositing the waste into appropriate bins. The machine learning model can accurately identify solid waste types based on materials like Wood, Metal, Paper, Plastic, and Glass. The aim of this research is to implement the model in both semi-urban and urban areas, enhancing the efficiency of waste segregation, with a willingness to modify physical components as needed.

I. Introduction

In today's world, garbage disposal has become a cause of major concern. An astounding amount of 0.1 million tons of waste is generated each day in India alone. Unfortunately, only 5% of this colossal amount of waste is recycled . This collection of garbage is then improperly disposed of in landfills without any segregation and also requires much more transportation facilities than necessary. This method of dumping causes adverse effects on the environment while failing to utilize the potential of the recyclable waste. Hazardous emissions and climate change are some effects caused by mishandling of waste. Despite the government's efforts to prioritize the separation of dry and wet waste, the norms are hardly followed and thus not effective. Moreover, any attempt to separate waste manually in small or large scale would require an extensive taskforce and countless hours of effort which would decelerate the process significantly.

In this work, a model is developed that can automatically segregate the waste without requiring any human intervention to tackle this problem. Machine learning techniques are very effective when it comes to classification and is used in the model to classify different types of waste. By using Image Processing algorithms: such as You Only Look Once version 3 (YOLOv3), the model can analyse the waste present in its workspace. Deep learning technique is used to train the model that can accurately isolate different types of waste based on the requirements. This would reduce the amount of garbage being transported and dumped in landfills while allowing maximum reusability. This solution can be ideally applied at primary levels i.e. where the wastes are produced. Raspberry Pi is used to control and segregate the waste. The proposed system aims to replace the manual process in waste segregation process by using image processing for waste detection and sorting based on machine learning algorithms.The system consumes minimal power and does not require complex hardware.

Background study:

The mainstream object detection algorithms are based on convolution neural networks (CNN), which are one-stage and two-stage detections, using different feature extraction methods. Object detection algorithms that adopt a two-stage detection method include regions with CNN (R-CNN), fast regions with CNN (Fast R)-CNN, and Faster R-CNN, which divide the detection task into region proposal and classification. Meanwhile, the one-stage detection method integrates region proposal and classification into one step, which reduces the detection time. The mainstream methods of one-stage detection are Single Shot Detector (SSD) and YOLO . In recent years, deep learning algorithms in computer vision is a rapidly developing research topic in classifying floating debris. The work in Fulton et al. classified three classes of trash, which are plastic debris, biological materials, and manmade objects. The work applied the Faster R-CNN method and resulted in 81 mAP of performance when classifying 820 test images. The trash detection system proposed in Fulton et al. applied the two-stage detection approach. Region proposal network (RPN), which is a fully convolutional network (FCN), predicts object boundaries and confidence scores at each point of location simultaneously. The RPN is aimed to produce high quality region proposals. RPN is combined with Fast R-CNN to form a network by sharing their convolutional features to produce Faster R-CNN. In Faster R-CNN, regions of interest are generated from the input image and these are transferred to subsequent convolutional layers. The RPN generates region proposals using the previously generated feature map. RPN adopted a sliding window over the feature maps while each window will generate k anchor boxes of different shapes and sizes. Then, adjacent pixels are clustered by texture, color, or intensity into the classifiers. After training, the classifiers on each region proposal will be returned for object detection purposes. Despite the high localization and recognition accuracy, a two-stage detection approach suffers from slow detection speed and is not applicable for real-time applications. On the other hand, a one-stage detection approach based on the YOLOv3 model is applied in the garbage detection system in Watanabe et al. and Li et al. . The work in Watanabe et al. demonstrated YOLOv3-based object detection for monitoring marine debris with an mAP of 77.2% by using 37 test images. Meanwhile, Li et al. developed a vision-based water surface garbage capture robot using a modified YOLOv3 model that is able to detect plastic bottles, plastic bags, and styrofoam. The performance of the model was evaluated based on 301 test images and achieved an mAP of 91.4%. The works in Watanabe et al. and Li et al. solved object detection as a regression problem that considers the whole image as input and instantaneously produces class probabilities and multiple bounding boxes. Hence, this has made the detection model much faster compared to

the two stages of object detectors. A You Only Look Once (YOLO) detector adopts the whole image as the network input, which will then be separated into an s × s grid. Then, the model will provide the position of the object border and the corresponding class in the output layer. The idea of the YOLO detector was to employ an exclusive neural network to the whole image, whereby the network splits the image into sections and concurrently predicted probabilities and bounding boxes for each section. The weight of the bounding boxes is computed based on the predicted probabilities. Each bounding box will have its

TABLE 1 The training and test datasets.

Object class	Training images	Test images
Plastic bottle	3,798	1,085
Aluminum can	2,799	586
Plastic bag	2,060	551
Styrofoam	487	146
Plastic container	410	113
Total	9,554	2,481

confidence score and the prediction is produced as a static number of boundary boxes. The prediction detects one object for each grid cell by applying a non-maxima suppression algorithm. YOLO usually adopts ImageNet to pre-train parameters and then applies a target detection dataset to recognize the training. Nevertheless, previous works focused on a small number of test images, which is that as more test images are considered, the classification process will become more complex. Hence, a more effective object detection algorithm is required to achieve good classification accuracy. Based on previous works reported in Junos and. Morera , SSD and YOLO detectors have provided feasible outcomes under various conditions with respect to image sizes, illumination, viewing perspectives, incomplete occlusion, and complex background. The benefit of applying the SSD model was the exclusion of False Positive (FP) cases. On the contrary, YOLO has shown to give better object localization results detecting a higher number of True Positive (TP) panels with higher accuracy. Moreover, YOLOv3 had shown to have significantly good precision and a real-time object detection algorithm that combined the features of YOLOv1, YOLOv2, and YOLOv3. In addition, YOLOv3 can achieve the existing optimum detection speed with a trade-off in detection accuracy. On the other hand, YOLOv3-tiny and YOLOv3-tiny are the lightest versions of the YOLOv3 model and YOLOv3 model, respectively. Although the YOLO-tiny models are simpler and less complex structures, the detection performance of the YOLO-tiny models was reduced significantly due to the weak feature extraction process

Proposed method

Image dataset and transfer learning

In this work, the training images are obtained under various conditions in terms of brightness and positions to prevent overfitting. Five classes of the floating debris database are developed, which include styrofoam, plastic bags, plastic bottles, plastic containers, and aluminum cans (Table 1). Pre-trained convolutional weights are applied for the training process to improve the accuracy of the object detector and lessen the computation time. Generally, applying inadequate learning data will produce inaccurate object detection performance. Thus, transfer learning is applied to aid the training process and obtain substantial results without having to include massive data. Hence, this work adopted transfer learning and applied the pretrained weights from Microsoft Common Objects in Context (MS-COCO) dataset to improve the model performance. The weights of the convolutional layers for the proposed model are pre-trained based on the MS-COCO dataset. The MS-COCO dataset is a large-scale image dataset that contains annotations that enable users to train the computer vision models to recognize, label, and describe objects. In addition, the MSCOCO dataset complements the transfer learning process where the data is applied for one model which serves as an initial point for another. The MS-COCO dataset is an important benchmark for computer vision to train, test,

and refine the object detection model. The formerly learned MS-COCO features provide the model with additional image recognition necessities required for the object detection process (38).

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β
Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

3. Model hyperparameters

_BATCH_NORM_DECAY = 0.9

_BATCH_NORM_EPSILON = 1e-05

_LEAKY_RELU = 0.1

_ANCHORS = [(10, 13), (16, 30), (33, 23),
(30, 61), (62, 45), (59, 119),
(116, 90), (156, 198), (373, 326)]

_MODEL_SIZE = (416, 416)

(_MODEL_SIZE refers to the input size of the model.)

Batch normalization

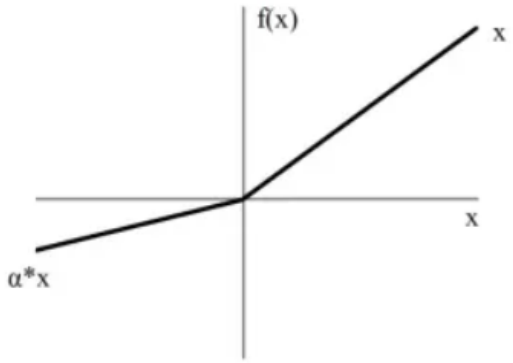
Almost every convolutional layer in Yolo has batch normalization after it. It helps the model train faster and reduces variance between units (and total variance as well). Batch normalization is defined as follows.

_BATCH_NORM_EPSILON refers to epsilon in this formula, whereas _BATCH_NORM_DECAY refers to momentum, which is used for computing moving average and variance. We use them in forward propagation during inference (after training).

moving_average = momentum * moving_average + (1 - momentum) * current_average

Leaky ReLU

Leaky ReLU is a slight modification of ReLU activation function. The idea behind Leaky ReLU is to prevent so-called "neuron dying" when a large number of activations become 0.



(_LEAKY_RELU refers to alpha.)

Anchors

Anchors are sort of bounding box priors, that were calculated on the COCO dataset using k-means clustering. We are going to predict the width and height of the box as offsets from cluster centroids. The center coordinates of the box relative to the location of filter application are predicted using a sigmoid function.

$$b_x = \sigma(t_x) + c_x$$
$$b_y = \sigma(t_y) + c_y$$
$$b_w = p_w e^{t_w}$$
$$b_h = p_h e^{t_h}$$

Where bx and by are the center coordinates of the box, bw and bh are the width and height of the box, cx and cy are the location of filter application and ti are predicted during regression.

4. Model definition

We referred to the official ResNet implementation in Tensorflow in terms of how to arrange the code.

Batch norm and fixed padding

It's useful to define batch_norm function since the model uses batch norms with shared parameters heavily. Also, same as ResNet, Yolo uses convolution with fixed padding, which means that padding is defined only by the size of the kernel.

Feature extraction: Darknet-53

For feature extraction Yolo uses Darknet-53 neural net pretrained on ImageNet. Same as ResNet, Darknet-53 has shortcut (residual) connections, which help information from earlier layers flow further. We omit the last 3 layers (Avgpool, Connected and Softmax) since we only need the features.

Convolution layers

Yolo has a large number of convolutional layers. It's useful to group them in blocks.

Detection layers

Yolo has 3 detection layers, that detect on 3 different scales using respective anchors. For each cell in the feature map the detection layer predicts n_anchors * (5 + n_classes) values using 1x1 convolution. For each scale we have n_anchors = 3. 5 + n_classes means that respectively to each of 3 anchors we are going to predict 4 coordinates of the box, its confidence score (the probability of containing an object) and class probabilities.

Upsample layer

In order to concatenate with shortcut outputs from Darknet-53 before applying detection on a different scale, we are going to upsample the feature map using nearest neighbor interpolation.

Non-max suppression

The model is going to produce a lot of boxes, so we need a way to discard the boxes with low confidence scores. Also, to avoid having multiple boxes for one object, we will discard

the boxes with high overlap as well using non-max suppression for each class.

Final model class

Finally, defined the model class using all of the layers described previously.

V. Results and Discussion:

The experiments in this work are executed using an epoch of 10,000 and the input image size of 416 × 416 × 3. First, this study investigates the contribution of transfer learning to the performance of the object detector by applying the MS-COCO dataset, in which the pre-trained convolutional weight is applied in the training stage. Figure 4 shows that the detection model with transfer learning provides a better converging rate compared to the model without transfer learning. The model that applied transfer learning continues to decrease its average loss until it achieves training steps of approximately 1,800 where the average loss converges to a constant level. This event shows a small value of decrement in the average loss until the termination of the training process. In the meantime, the model without transfer learning suffered from overfitting at about 3,700 training. This means that the detection model does not generalize well without including transfer learning. As a result, the overfitted model will not be able to perform well on the new test dataset which will affect the accuracy of the detection system. After some time, the average loss for both models constantly decreases but the model without transfer learning has a higher average loss until the end of the training process. Hence, this shows that the application of the transfer learning process will contribute to lower generalization error during the training process which will aid the detection process on the new test dataset

To determine the optimized network structure of the proposed model, the subdivision parameter is varied between 8 and 64 while keeping other parameters constant. Subdivision simply means the batch is split by the value of subdivision into mini batches. For example, for batch=64 and subdivision=8, the training will have 64/8 = 8 images per mini batches. These mini batches will then be sent to the GPU for the computation process. The computation process will be repeated 8 times until a batch is completed. The new iteration begins to be 8 in this work because it produces the lowest average loss during the training process. Averaging over more images helps to speed up the time for training and to generalize the training even more. However, this can be a problem when the subdivision is reduced because a memory issue might occur if the GPU does not have enough memory to process more images at one time.

Conclusion

In conclusion, an automated detection system based on the optimized YOLO model is developed to detect floating solid wastes that include plastic bottles, aluminum cans, plastic bags, styrofoam, and plastic containers. In this work, the proposed model optimized the network structure of the conventional YOLOv3 model that includes (i)

modification of CSPDark-Net53 into the backbone to overcome limitations due to training time, (ii) adoption of Hard-Swish activation function, and (iii) improved PANet in the Neck module to aid the feature extraction process. The performance of the proposed YOLO model is compared with previous works and has shown promising results with an mAP value of 89%. This research demonstrates that the computer vision system plays an important role in environmental monitoring and provides novel insights for improved decision-making and sustainable management. In a nutshell, this study is important for riverine management in urban landscapes since the river is an important part of urban ecological civilization and human health.