International Journal of Scientific & Engineering Research Volume 10, Issue 4, April-2019
ISSN 2229-5518

Artificial Neural Network based Intelligent Waste Segregator

Nitha Elizabeth John, Sreelakshmi R, Swetha R Menon, Varsha Santhosh

Rajagiri School of Engineering and Technology,Kakkanad,Kochi

Abstract

In today's fast-paced world, we are facing an escalating problem in ensuring efficacious and sustainable management of waste. This is a result of rapid increase in urbanization and industrialization. India ranks second in the world after China, in terms of population and this rising population has given way to an increase in the use of resources and ultimately resulting in a proportional increase in waste generation. Waste management has gained popularity as an issue that requires immediate attention and action. Waste segregation is the most important step in this process. The objective of this project is to capture images of a single waste material and effectively identify and segregate it into six classes viz. Metal, Cardboard, Glass, paper, plastic and miscellaneous. The proposed system exploits Deep learning and is mechanized using a robotic manipulator to ensure effective automated waste management and will speed up the process of segregation without any human intervention.

Index Terms - Deep learning, Artificial Neural Networks, Robotic Manipulator, and Waste Segregation

1. INTRODUCTION

1.1 Problem Definition

The existing waste segregation methods lack self-learning capability, are slow and inaccurate, hence requires constant replacement. They can work well in a small scale, but for large scale they are not very effective. It will be difficult to maintain these machines in large scale. Due to large amounts of waste generated, large number of such machines has to be bought. But buying and maintaining these machines will prove to be excessively costly. Also the accuracy of these methods are very low, especially the manual method which is prone to a lot of errors.

Making the current waste segregation system autonomous is a better solution to this matter by automatically sorting waste. Classifying wastes into recycling categories using Artificial Neural Networks can prove to be a very efficient methodology to process wastes.

1.2 Objective

The waste segregator system aims at automating the segregation using recent tools of Machine Learning.

- With the help of a powerful deep layered network called Convolution Neural Network, a Machine Learning tool, high accuracy of segregation results can be achieved.

International Journal Of Scientific & Engineering Research, Volume 10, Issue 4, April 2019 Edition
ISSN 2229-5518

- The network is trained for the various waste categories using the appropriate datasets.
- This trained network is integrated with a mechanical system that performs the physical segregation of waste, thereby avoiding human intervention.

2. DEEP CONVOLUTIONAL NEURAL NETWORK

Convolutional neural networks are deep artificial neural networks that are used primarily to classify images, cluster them by similarity, and perform object recognition within scenes. Convolutional neural networks ingest and process images as tensors, and tensors are matrices of numbers with additional dimensions and perform a sort of search. Convolutional networks perceive images as volumes; i.e. three-dimensional objects.

A convolutional network takes in square patches of pixels and passes them through a filter known as Kernel. The next layer in a convolutional network executes max pooling. The activation maps are fed into this layer, and like convolutions, this method is applied one patch at a time. Finally, a fully connected layer is used to classify the output.

3. DESIGN AND METHODOLOGY

3.1Overview

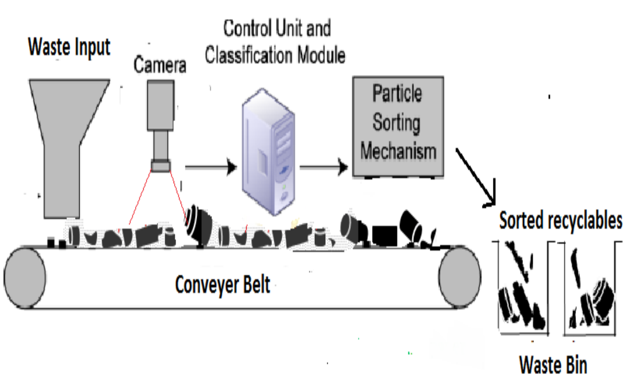

Figure1: Schematic Representation

The above given figure [2] is a schematic representation of the various processes that are undergone in the proposed project, from capturing the images to sorting different types of waste into their respective bins. Initially, the different types of waste materials are passed through the conveyer belt. The camera is positioned above the conveyor belt records the images. These images that are captured are given into the neural network (Convolution Neural Network is used). The completely trained ANN is in python language is converted into a hardware descriptive language (VHDL) before embedding into a portable processor, Raspberry Pi. This network which is pre-trained will categorize each waste into their respective or most probable categories as per the training. Based on this classification, these waste materials will be sent to their respective bins.

3.2 Methodology

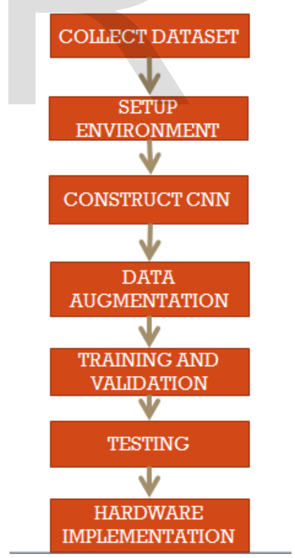

Figure 2: Flow chart illustrating the Methodology

3.2.1. Data set acquisition:

In order to train the neural network, we require huge amount of data only then the neural network can be trained efficiently. Higher the number of inputs, higher is the accuracy and vice-versa.

International Journal of Scientific & Engineering Research Volume 10, Issue 4, April-2019
ISSN 2229-5518

International Journal Of Scientific & Engineering Research, Volume 10, Issue 4, April 2019 Edition
ISSN 2229-5518

3.2.2. Data Augmentation

It adds value to base data by adding information derived from internal and external sources within an enterprise. Basically what we do is for a same input image we do translation, rotation and scaling since it is not practically possible to take large number of images.

3.2.3. Setting up of environment

Using Anaconda, virtual environments can be created, exported, listed, removed and updated that have different versions of python and /or packages like numpy, sklearn, keras with tensor flow backend installed in them [1]. Switching or moving between environments is called activating the environment. Creating the environment with the necessary package is the first step of programming in anaconda.

3.2.4. CNN Construction

A convolutional neural network (CNN, or ConvNet) is a class of deep, feed-forward artificial neural networks, most commonly applied to analyzing visual imagery [5]. It is a 3 layered network that consists of:

- Convolution layer-The convolutional layer is the core building block of a CNN. The layer's parameters consist of a set of learnable filters (or kernels), which have a small receptive field, but extend through the full depth of the input volume. During the forward pass, each filter is convolved across the width and height of the input volume, computing the dot product between the entries of the filter and the input and producing a 2-dimensional activation map of that filter. As a result, the network learns filters that activate when it detects some specific type of feature at some spatial position in the input.
- Pooling layer - Pooling is a form of non-linear down-sampling. There are several non-linear functions to implement pooling among which max pooling is the most common. It partitions the input image into a set of non-overlapping rectangles and, for each such sub-region, outputs the maximum [5]. The pooling layer serves to progressively reduce the spatial size of the representation, to reduce the number of parameters and amount of computation in the network, and hence to also control over fitting.
- Fully Connected Layer - After several convolution and max pooling layers, the high-level reasoning in the neural network is done via fully connected layers. Neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular neural networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset.

3.2.5. Training and Validation

The model is trained on the training dataset using a supervised learning method. The training dataset often consist of pairs of an input vector and the corresponding output vector, which is commonly denoted as the target. The current model is run with the training dataset and produces a result, which is then compared with the target, for each input vector in the training dataset. Based on the result of the comparison and the specific learning algorithm being used, the parameters of the model are adjusted. The model fitting can include both variable selection and parameter estimation. The fitted model is used to predict the responses for the observations in a second dataset called the validation dataset. The validation dataset provides an unbiased evaluation of a model fit on the training dataset

3.2.6. Testing

Testing data is used to test the system. It is the set of data which is used to verify whether the system is producing the correct output after being trained or not. Testing data is used to measure the accuracy of the system.

3.2.7. Hardware Implantation

The trained classifier is implemented on a raspberry pi which is coupled with a robotic actuator to pick and place the materials from the conveyor belt to the respective bins.

3. RESULT

3.1 Observation

As the number of images being trained is increasing, the loss is decreasing and the accuracy is improving steadily. The completion of an epoch updates weights. The training was complete with an accuracy of 80%. The entire training stage results in a fully trained neural network based classifier which can classify the images captured into 6 classes, metal, cardboard, plastic, paper, glass and trash. This ANN is then effectively embedded into a Raspberry Pi board. This effective classification method can be extended to many categories and using Raspberry Pi makes it a portable ANN and can be utilized to build an autonomous waste segregation system which is the goal of the proposed project.

Following is the output obtained from the simulation of a 2class model (i.e. only metal and cardboard taken into consideration).

```
Using TensorFlow backend.
C:\Users\user\Anaconda3\envs\tensorflow\lib\site-packages\ipykernel_launcher.py:29: UserWarning: Update your `Dense`
call to the Keras 2 API: `Dense(units=256, activation="relu")`
C:\Users\user\Anaconda3\envs\tensorflow\lib\site-packages\ipykernel_launcher.py:30: UserWarning: Update your `Dense`
call to the Keras 2 API: `Dense(units=128, activation="relu")`
C:\Users\user\Anaconda3\envs\tensorflow\lib\site-packages\ipykernel_launcher.py:31: UserWarning: Update your `Dense`
call to the Keras 2 API: `Dense(units=1, activation="sigmoid")`
Found 940 images belonging to 2 classes.
Found 459 images belonging to 2 classes.
Epoch 1/15
 97/950 [==>...........................] - ETA: 2:09:06 - loss: 0.7841 - acc: 0.7745
```

Figure 3: After training the 97th image, the accuracy is 0.7745

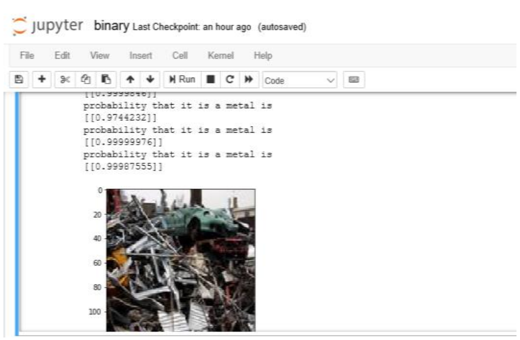

Figure 4: Depicts the different probabilities that the given test image is a metal in different epochs

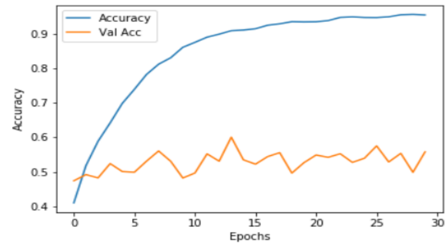

Figure 5: Graphs showing accuracy and loss for different epochs

4. FUTURE EXPANSION

1. Use of hyper spectral images to expand the scope of the proposed project so that it can be utilized in differentiating the materials at an elemental level.

2. Implementation of the trained classifier onto an FPGA.

5. REFRENCES

[1] François Chollet.2017. "Deep learning with PYTHON" Manning Publications Company

[2] Amos R. Omondi and Jagath C. Rajapakse.2006. "FPGA Implementations of Neural Networks "published by Springer

[3] Ali Mohammad Tehrani and Hamidreza Karbasi.2017. "A Novel Integration of Hyper-Spectral Imaging and Neural Networks to Process Waste Electrical and Electronic Plastics"- 2017 IEEE Conference on Technologies for Sustainability

[4] RomanA. Solovyev, Alexandr A. Kalinin, Alexander G. Kustov, Dmitry V. Telpukhov and Vladimir S. Ruhlov. FPGA Implementation of Convolution Neural Networks with a Fixed-Point calculation.

[5] K.he,Xhang,S.Ren and J.Sun.2016 "Deep residual learning for image recognition", Proceedings of the IEEE conference on computer vision and pattern recognition.