**Convolutional Neural Networks Application in Plastic Waste Recognition and Sorting**

Andrey N. Kokoulin1, Aleksandr I. Tur,

Aleksandr A. Yuzhakov

Electrotechnical department

Perm National Research Polytechnic University

Perm, Russia

1a.n.kokoulin@ieee.org

*Abstract*— In this paper authors describe their results in IoT and Convolutional Neural Networks application in reverse vending machine project. In Europe and United States a reverse vending machine is a device that accepts used (empty) beverage containers and returns money to the user. In Russia we can provide the concept of automatic processing of plastic and metal waste combined with another mechanism for economic motivation of end users – system of discounts and bonuses in large trading networks. These machines have to be as cheap as possible but with rich functionality and security. Traditionally the basics of this machine is assembled with IoT controllers and tiny single-board computers which have heavy memory and computational restrictions. Nevertheless these controllers are able to recognize the waste types using cameras and provide sorting and some kind of preprocessing. We show our CNN implementation on IoT for reverse vending machine.

Keywords—reverse vending machine; IoT; convolutional neural network; histogram of oriented gradients

# Introduction

A reverse vending machine (RVM) is a machine where people can return empty beverage containers like bottles and cans for recycling. The machine often gives back a deposit or refund amount to the end user. This is what makes it a “reverse” vending machine: instead of the user putting in money and getting out a product (like at a candy vending machine), the user puts a product **in**and gets **out**a monetary value. Reverse vending systems are an automated way to collect, sort and handle the return of used drink containers. Reverse vending machines are especially common in regions with container deposit laws (where you get money back for returning certain containers) or mandatory recycling legislation. The machines recognize each material and compact then separates the materials, enabling the materials to be “recycling” ready with no contaminates. The value of the collected materials is increased and the volume reduced, enabling reduced logistics costs. Refillable containers are transported back to the bottle or beverage producer for cleaning and refilling. Non-refillable containers are taken to a processing facility for washing and shredding, to go into the production of new containers. The materials must meet regulations for quality and hygiene in order to be used as new drink containers. Reverse vending machines are a key part of container deposit systems in Europe and United States, which takes 70% to almost 100% of all drink containers returned for recycling [1].

As for the issue of preventing fraud, the system is easily established through the application of multiple security checks performed by RVMs. Basically it is a simple issue of how advanced the RVM is, i.e. what control function it can perform.

For instance, RVM machines do it as follows:

* control of the material of the container (e.g. by IR-spectrometer)
* control of the shape of the container
* control of the barcode

These three basic control-procedures prevent fraud or attempted fraud and make any attempt of the fraud completely impossible. But the same time it makes the RVM too expensive. For example Capital Purchase prices for reVend 800 Reverse Vending Recycling Machine would be [2]:

1 machine = £7,000 + VAT



Fig 1. Example of RVM exterior.

With the modern computer vision technologies we can design another kind of efficient and non-expensive RVM having the same functionality and high fraud-protection. This is actual for Russian reality because we have no container deposit laws or something equal. But we can provide the concept of automatic processing of plastic and metal waste combined with another mechanism for economic motivation of end users – system of discounts and bonuses in large trading networks. So each recycled container can give the non-monetary opportunity for user. Also the profit of RVM owner is measured by the cost of recycled plastic or metal and is very low with great logistic charges, making the RVM invention and mantaining issues actual for local governments.

# Reverse Vending Machine design

The recycler places the empty bottle or can into the receiving aperture; the horizontal in-feed system allows the user to insert containers one at a time. In RVMs the bottle/can is scanned and identified (matched to database) and determined to be a participating container. Foreign machines can use material recognition instead of/as well as a bar code scanner when needed along with some online or local database of barcodes. This leads to RVM price increase because of IR-spectrometer and bar code scanner and accompanying electronics expenses. It also makes the recognition process more complicated due to databases daily synchronization to process new barcodes for new types of containers from thousands of vendors. We tried to implement another identification method based on object recognition with neural networks (NN) using tiny IoT controller device (Raspberry PI 3) with screen, camera and some sensors and servos, spending less than $100 for control system. This approach does not depend on sample database renewals; NN studies the different forms of cans and PET bottles and identifies the main features needed for classification. Due to this specialty any new PET bottle image could be identified as aggregate class of PET bottles if the container possesses the corresponding features. The can is processed the same way and could be identified as aggregate class of cans even if container is jammed and twisted. Also we train the NN to detect fraud (including glass bottles), hands, and fire in the aperture to avoid the transportation servo switching on. Thus the system is autonomous and is able to process any type of containers.

Container is removed from aperture into corresponding basket using servo. In one our RVM model one-time-use container is pinned and crushed to reduce its size, to avoid spillages of liquid and to increase storage capacity.

RVM is equipped with feedback system to notify the logistic center about filling of receiver and some kinds of security events.

After container identification and processing by RVM the user is offered a choice of bonuses listed on the screen. The choice is transformed into the QR-code which can be scanned by user’s smartphone via corresponding application. So the upper level operations are delegated to the smartphone including the user authentication, logging into the trading network web-portal for bonuses requesting and so on. Thus we separate and distribute the functionality and reduce user-RVM interaction. The main advantages are information security of user authentication and data exchange, independence from GPRS/3G connections for RVM, absence of inconvenient sensor or physical keyboards on RVM’s front panel.

# Neural Network for Waste Objects Recognition

Deep learning approach is the most advanced level of neural networks. Modern researchers can obtain powerful specialized hardware with more training data to perform machine learning and thus have more abilities to train networks with dozens of hidden layers that are capable of hierarchical learning where simple concepts are learned in the lower levels and more complicated abstract patterns in the higher layers of the network. Convolutional neural networks (CNN) which automatically learns discriminative patterns (filters) from images by sequentially stacking convolutional layers on top of each other are the most effective NNs. In many applications CNNs are considered the most powerful image classifier and are currently responsible for pushing the state-of-the-art forward in computer vision subfields that leverage machine learning [4,5].

Machine learning algorithms are divided into three types:

* Supervised;
* Unsupervised;
* Semi- supervised.

In the supervised case a CNN is given both a set of inputs and target outputs. The algorithm then tries to learn patterns that can be used to automatically map input data to their correct target output: machine learning algorithm tries to guess the correct answer and if it fails, the tuitor guides it toward a better and precise guess at next time. So the CNN aim in image classification is to take sets of images and identify patterns that can be used to discriminate various images classes (object’s shapes) from one another.

In our paper we consider some approaches in computer vision and image processing and their application to the problem of automatic recognition of empty recyclable containers (bottles and cans) and detecting fraud. The NN is designed as the Python script, which is loaded after the sensor has fired event and camera has made shot. The two main parameters for NN script are path to model file and path to the image to be processed.

The Python feature package contains many methods to extract features from images. In order to detect objects in OpenCV and remove the false positives the common practice is tuning the  cv2.detectMultiScale() function  parameters. But there is no guarantee that the exact same parameters will work from image-to-image. This makes batch-processing of images for object detection a tedious task since we’ll be very concerned with either (1) falsely detecting objects or (2) missing objects entirely, simply due to poor parameter choices on a per image basis. This happens because this function is based on the fast Haar cascade classifiers provided by OpenCV (i.e. the Viola-Jones detectors) and we have to check another solutions for object detection which show the stable positive results and are portable and convenient to use at the same time. We can use such object detection frameworks as Local Invariant Descriptors, Histogram of Oriented Gradients, Deformable Parts Models, Exemplar Models utilizing Deep Learning methods [3,4].

We have to restrict the list of the available methods and frameworks because of IoT controllers and tiny single-board computers we use have memory and computational restrictions. We cannot apply all the possible neural network and we decided to use some of the most popular networks based on *Keras* over *Tensorflow* and *Caffe* frameworks. They are:

* LeNet;
* AlexNet;
* SqueezeNet.

But also we have our first implementation of object detection algorithms based on Histogram of Oriented Gradients (HOG) image descriptor and a Linear Support Vector Machine (SVM) and we should start with this method.

We should note that our aim is to classify the image inside the RVM by three possible classes: PET bottle, aluminum can or fraud (everything that doesn’t match PET bottle or can). We take into attention that those cans or bottles could be twisted or jammed and we included corresponding images into training and test sets.

## Histogram of Oriented Gradients

Similar to edge orientation histograms and local invariant descriptors such as SIFT, HOG operates on the gradient magnitude of the image. The main idea of ​​the algorithm is the assumption that the appearance and shape of the object in the image area can be described by the distribution of intensity gradients or the direction of the edges. However, unlike SIFT, which computes a histogram over the orientation of the edges in small, localized areas of the image, HOG computes these histograms on a dense grid of uniformly-spaced cells. Furthermore, these cells can also overlap and be contrast normalized to improve the accuracy of the descriptor.

HOG has been used successfully in many areas of computer vision and machine learning, but especially noteworthy is the detection of people in images (if they have the same vertical position). In this case, we are going to apply the HOG image descriptor and a Linear Support Vector Machine (SVM) to learn the representation of image bottles and cans. We’ve got into assumption that images of containers have to be similarly oriented and also similarly scaled.

But unexpectedly this method has shown minor results on our datasets even after several tuning attempts. It seems to us that this is the method’s peculiarity - we used the 2-D image stretched into a 1-D vector as a descriptor. So the cylindrical shape of bottles and cans could result in similar values of gradients for repeating long run-length sequences in descriptor. And the SVM method has a significant drawback - even one erroneous example can spoil the entire class, becoming a reference vector. So the amount of false positive and false negative results was unacceptable.

## LeNet network

Another approach was LeNet based on Keras and Tensorflow frameworks. This network automatically learns discriminative patterns (filters) from images by sequentially stacking convolutional layers on top of each other and gain more precise recognition accuracy [6].

1. Results of LeNet Testing

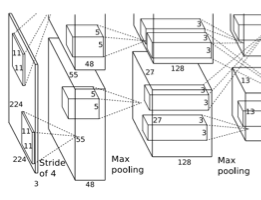
|  |  |  |  |
| --- | --- | --- | --- |
| NN type: | | LeNet - 6 classes | |
| Model name: | | model\_6\_28.model | |
|  | | Correctness | % |
| PET | #1 | 1 | 91 |
| #2 | 1 | 92 |
| #3 | 1 | 85 |
| #4 | 1 | 86 |
| #5 | 1 | 91 |
| cans | #6 | 0 | 82 |
| #7 | 0 | 67 |
| #8 | 1 | 63 |
| #9 | 1 | 69 |
| #10 | 0 | 53 |
| other | #11 | 1 | 55 |
| #12 | 1 | 83 |
| #13 | 1 | 98 |
| #14 | 1 | 93 |
| #15 | 1 | 98 |
| Correctness | | 80% | |

Each input image in learning set is resized into 28x28 pixels image and then passed through several hidden layers. So, the feature vector is of length 784 (28\*28). This is amount of nodes in input layer.

1. Results of LeNet Testing with 2 Models

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NN type: | | LeNet - 2класса | | | |
| Model name: | | саn\_sharpen.model | | pet\_56.model | |
|  | | Correctness | % | Correctness | % |
| PET | #1 | 1 | 98 | 1 | 91 |
| #2 | 1 | 96 | 1 | 76 |
| #3 | 1 | 96 | 1 | 69 |
| #4 | 1 | 91 | 1 | 67 |
| #5 | 1 | 58 | 1 | 99 |
| cans | #6 | 0 | 84 | 0 | 70 |
| #7 | 0 | 99 | 1 | 92 |
| #8 | 1 | 99 | 1 | 64 |
| #9 | 1 | 88 | 1 | 88 |
| #10 | 1 | 74 | 1 | 58 |
| other | #11 | 1 | 72 | 1 | 98 |
| #12 | 1 | 99 | 1 | 99 |
| #13 | 1 | 96 | 1 | 98 |
| #14 | 1 | 99 | 1 | 98 |
| #15 | 1 | 99 | 1 | 98 |
| Correctness | | 87% | | 93% | |

Fig 2. AlexNet Inpul Layer and First Convolutional Layers.



Hidden layers are the unsupervised Restricted Boltzmann Machine where the output of each RBM in the hidden layer is used as input to the next [6].

The visible output layer contains the output probabilities for each class label. The output node which produces the largest probability is chosen as the overall classification. We can sort the output probabilities and choose all class labels that lay within some range near the largest probability – so we can find most likely class labels and thus perform soft-decisions.

We have prepared several types of models and tested them on several containers images. The results are shown in Table 1.

One of the models contained 6 classes:

* PET bottles
* Cans
* Glass bottles
* Fired fraud
* Man’s hand
* Other fraud

We supposed that the recognition accuracy should be better if we manually divide all possible objects into classes that are as diverse as possible.

From another point of view we can try to create paired models for LeNet CNN, each of them is trained on two classes: “PET bottles – not PET bottles” and “cans – not cans”. We can start two Python scripts for corresponding models simultaneously and receive some results from output layers. This kind of recognition processing perfectly suites our goals: the designed RVM has two apertures for PET bottles and cans respectively. If we put the PET in the can’s aperture it has to be recognized as unacceptable fraud and needs to be removed.

This double LeNet application shows better results than 6-classes LeNet on the same testing samples (Table II).

## AlexNet network

AlexNet is more complicated CNN than LeNet (Fig.2). The training images are resized to 224x224 samples and passed to hidden layers. This provides the more detailed patterns studying than 28x28 samples in LeNet. AlexNet is the neural network, which has 60 million parameters and 500,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and two globally connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of convolutional nets. To reduce overfitting in the globally connected layers we employed a new regularization method that proved to be very effective [5,7]. The first convolutional layer filters the 224×224×3 input image with 96 kernels of size 11×11×3 with a stride of 4 pixels (this is the distance between the receptive field centers of neighboring neurons in a kernel map). The second convolutional layer takes as input the (response-normalized and pooled) output of the first convolutional layer and filters it with 256 kernels of size 5 × 5 × 48. The third, fourth, and fifth convolutional layers are connected to one another without any intervening pooling or normalization layers. The third convolutional layer has 384 kernels of size 3 × 3 × 256 connected to the (normalized, pooled) outputs of the second convolutional layer. The fourth convolutional layer has 384 kernels of size 3 × 3 × 192 , and the fifth convolutional layer has 256 kernels of size 3 × 3 × 192. The fully-connected layers have 4096 neurons each.

It is evidently that this CNN provides better accuracy than LeNet with comparable recognition speed on Raspberry PI 3 (see Table III).

1. Results of AlexNet Testing

|  |  |  |  |
| --- | --- | --- | --- |
| NN type: | | AlexNet | |
| Model name | | bottles\_another.model | |
|  | | Correctness | % |
| PET | #1 | 1 | 91 |
| #2 | 1 | 92 |
| #3 | 1 | 85 |
| #4 | 1 | 86 |
| #5 | 1 | 91 |
| cans | #6 | 0 | 82 |
| #7 | 1 | 69 |
| #8 | 1 | 63 |
| #9 | 1 | 69 |
| #10 | 1 | 53 |
| other | #11 | 1 | 55 |
| #12 | 1 | 83 |
| #13 | 1 | 98 |
| #14 | 1 | 93 |
| #15 | 1 | 98 |
| Correctness | | 93% | |

The time delay of training is not considered here, because we were training our model on modern PC with GPU CUDA abilities.

## SqueezeNet network

Recent research on deep neural networks has focused primarily on improving accuracy. For a given accuracy level, it is typically possible to identify multiple DNN architectures that achieve that accuracy level.

1. Results of SqueezeNet Testing

|  |  |  |  |
| --- | --- | --- | --- |
| NN type: | | SqueezeNet | |
| Model name: | | bottles\_another.model | |
|  | | Correctness | % |
| PET | #1 | 1 | 100 |
| #2 | 1 | 100 |
| #3 | 1 | 100 |
| #4 | 1 | 100 |
| #5 | 1 | 100 |
| cans | #6 | 0 | 95 |
| #7 | 0 | 100 |
| #8 | 1 | 100 |
| #9 | 1 | 100 |
| #10 | 1 | 100 |
| other | #11 | 1 | 100 |
| #12 | 1 | 100 |
| #13 | 1 | 100 |
| #14 | 1 | 100 |
| #15 | 1 | 100 |
| Correctness | | 87% | |

With equivalent accuracy, smaller DNN architectures offer at least three advantages:

* Smaller DNNs require less communication across servers during distributed training.
* Smaller DNNs require less bandwidth to export a new model from the cloud to an autonomous car.
* Smaller DNNs are more feasible to deploy on FPGAs and other hardware with limited memory.

To provide all of these advantages, we propose a small DNN architecture called SqueezeNet[5]. SqueezeNet achieves AlexNet-level accuracy on ImageNet with 50x fewer parameters. Additionally, with model compression techniques we are able to compress SqueezeNet to less than 0.5MB (510x smaller than AlexNet).We have got less precious results for the same testing set but our model was less than 0.5MB size and the recognition time was two times faster. In the RVM case the delay and memory consumption are not the decisive factor, but in some other IoT applications this could be important.

# Conclusion

The new result is estimation of several approaches in image recognition and classification and their application to empty containers recognition and sorting in Reverse Vending Machine. The most accurate decision was processed by AlexNet CNN, but we have obtained nearly the same accuracy with LeNet models after several enhancements.

This research was not limited to studying of CNNs porting to IoT devices possibility. We tried to understand how image preprocessing and enhancing of training sets and testing sets could influence the recognition accuracy and after several experiments found out the following for container recognition:

* LeNet training could be more efficient if we use 56x56 and higher resizing of training images;
* Use of double LeNet models each of 2 classes is more efficient than single model of 3-6 classes;
* Training images set preprocessing (sharping) can enhance recognition only for cans 2-classes model but decrease accuracy for PET bottles 2-classes model. So we have to tune these two models in a different ways;

##### references

1. Reverse vending 101: a beginner’s guide (2011). Available at: <https://www.tomra.com/en/collection/reverse-vending/reverse-vending-news/2017/how-does-a-reverse-vending-machine-work/> (Accessed 30 December 2017)
2. Cost of a Deposit Return Reverse Vending Machine. Available at: [http://www.zerowastescotland.org.uk/sites/default/files/reverse%20v %20-%20CFE%20response.pdf](http://www.zerowastescotland.org.uk/sites/default/files/reverse%20v%20%20-%20CFE%20response.pdf) (Accessed 30 December 2017)
3. Rosebrock A. Histogram of Oriented Gradients and Object Detection. Available at: <https://www.pyimagesearch.com/2014/11/10/histogram-oriented-gradients-object-detection/> (Accessed 30 December 2017)
4. Rosebrock A. Practical Python and OpenCV + Case Studies, 2016. Available at: <https://www.pyimagesearch.com/2016/11/20/Python-OpenCV/> (Accessed 30 December 2017)
5. Iandola F. Forrest, Han Song, Moskewisz W. Mattew et. al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 Mb model size, ICLR’17 conference proceedings, pp.207-212, 2017.
6. LeCun Y., Bottou L., Haffner P., Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278-2324. 1998.
7. Krizhevsky Alex, Sutskever Ilya, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks Available at: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf> (Accessed 30 December 2017)