International Journal of Scientific Research & Engineering Trends
Volume 6, Issue 4, July-Aug-2020, ISSN (Online): 2395-566X

Packing Automation in a High Variety Conveyor line via Image Classification

B. Arjun, Vidhu Hari, Dileep Chandran, Assistant Professor Ajay Basil Varghese
Department of Computer Science and Engineering
Adi Shankara Institute of Engineering and Technology, Kalady, Ernakulam.Kerla, India
barjun2009@gmail.com, vidhuvvsahari@gmail.com, dileepchandranc.dc.dc@gmail.com

***Abstract*-Due to the rise in computational power available, there has been a steady increase of interest in fields such as image recognition and machine learning. Consequently, one of the best areas for its application is in the field of factory automation since it requires processing of large amounts of image data within strict time constraints. Implementing such systems has become much easier due to the rise of machine learning frameworks such as Tensorflow. Hence implementing a Machine learning based inference model to do routine tasks such as object identification and verification can greatly increase the overall efficiency and offer great monetary returns by minimizing human labor. An SSD Mobilenet based model was used for this purpose for its speed and accuracy.**

***Key words*-image recognition, machine learning, Tensorflow, inference model, automation, SSDMobilenet**

I.INTRODUCTION

Today, technology is shaping every part of our world-hence every aspect of life should be adaptable , be it the business field,education or the field of industrial manufacturing. Having access to well functioning and high technology equipment is very necessary in the process of creating high quality products or services. This is one of the reasons why automation in factories is becoming hugely popular within the last decade or so.

Automation is generally accepted as the process of producing a certain product in a more precise and efficient manner. It consists of using technologies such as high speed cameras and identification systems used to efficiently channel the flow of goods within an industrial environment. Cost effective and highly capable vision systems running robustsoftware are transforming longstanding autonomous and adaptive industrial automation aspirations into reality.

Automated systems in manufacturing line environments are capable of working more tirelessly, faster and more exactly than humans. Embedded vision innovations can help improve product tracking through production lines and with enhanced storage efficiency. While barcodes and radio-frequency identification tags can also help track and route materials, they cannot be used to detect damaged or flawed goods.

Intelligent raw material and product tracking and handling in the era of embedded vision will be the foundation for the next generation of inventory management systems, as image sensor technologies continue to mature and as other vision processing components become increasingly integrated. High-resolution cameras can already provide detailed images of work material and inventory tags, but complex, real-time software is needed to analyze the images, to identify objects within them, to identify ID tags associated with these objects, and to perform quality checks.

II. LITERATURE REVIEW

R. Visalatchi, T. Navasri, P. Ranjanipriya and R. Yogamathi, "Intelligent Vision with TensorFlow using Neural Network Algorithms," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2020, pp. 944-948, doi: 10.1109/ICCMC48092.2020.ICCMC-000175.
The authors of the project compared the speed and accuracy of both YOLOv3 based and SSD-Mobilenet based architectures on an object detection problem.The performance of the models under varying conditions of lighting and complex changing environments.The end result was that the SSD based model consistently provided better accuracy whilesacrificing negligible amounts of speed against the YOLO classifier.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu: “SSD: Single Shot MultiBox Detector”, 2016; arXiv:1512.02325.
This is the foundational paper on the SSD network.The authors presented a new type of CNN that goes through the images only once rather than having separate region proposal and identification stages such as in RCNN based approaches.They theorized that with sufficiently powerful hardware it was possible to perform real time object detection with this algorithm.

International Journal of Scientific Research & Engineering Trends
Volume 6, Issue 4, July-Aug-2020, ISSN (Online): 2395-566X

J. Jia, "A Machine Vision Application for Industrial Assembly Inspection," 2009 Second International Conference on Machine Vision, Dubai, 2009, pp. 172-176, doi: 10.1109/ICMV.2009.51.
The authors performed a comparison between various off the shelf embedded camera based machine vision units.They used classical machine vision methods and hence the model tends to be rule based, i.e, cannot perform accurately when the input has too much noise.This is due to the fact that machine learning systems were not popular at that time.

Pouramini and H. Varaee, "A machine vision system for defect detection of a traveling grate conveyor," 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, 2015, pp. 1063-1066, doi: 10.1109/KBEI.2015.74361
In this paper, an online system for automatic defect detection of grates of a travelling grate conveyor was presented. After capturing the video, its frames are converted to binary images.A sliding window over the grate measures the black pixels ratio in each window and the damaged parts are detected.The detection rate was found to be more than 98% in test results

Li, Yiting& Huang, Haisong&Xie, Qingsheng& Yao, Liguo& Chen, Qipeng. (2018). Research on a Surface Defect Detection Algorithm Based on MobileNet-SSD. Applied Sciences. 8. 1678. 10.3390/app8091678.
This paper aims to achieve real-time and accurate detection of surface defects by using a deep learning method. For this purpose, the Single Shot MultiBox Detector (SSD) network was adopted as the meta structure and combined with the base convolution neural network (CNN) MobileNet into the MobileNet-SSD. Then, a detection method for surface defects was proposed based on the MobileNet-SSD.

Specifically, the structure of the SSD was optimized without sacrificing its accuracy, and the network structure and parameters were adjusted to streamline the detection model. The proposed method was applied to the detection of typical defects like breaches, dents, burrs and abrasions on the sealing surface of a container in the filling line. The results show thatour method can automatically detect surface defects more accurately and rapidly than lightweight network methods and traditional machine learning methods. The research results shed new light on defect detection in actual industrial scenarios.

III. DATASET AND DETECTION MODEL

1. Footwear Data-set

The presented model was trained using a data-set consisting of 1233 images. Each of these images were captured using a smartphone camera in burst mode. This ensures all round coverage of the footwear.

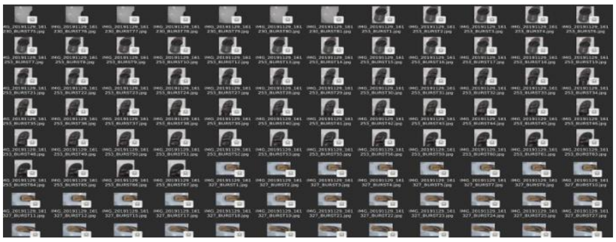

Fig 1 Data Set.

All the images were taken against a white background. Once the images were acquired, the region of the image containing the

Fig 2 Cropping.

footwear was cropped out by subtracting the region containing the white background using a Python script. These cropped images were then merged with random images from another data-set so as to create pictures with varying backgrounds. This is done so as to improve the final accuracy of the trained model. Since SSD based networks require a large amount of data to generate accurate predictions, we used various data augmentation methods to rotate, crop and change various characteristics of the images to create a varied database.

2. SSD-MobileNetV2 Classifier

Our proposed model is based on the SSD-MobileNetV2 architecture [1].The MobileNet network was developed to improve the real-time performance of deep learning under limited hardware conditions.The network can reduce the number of parameters without sacrificing accuracy.Previous studies have shown that the MobileNet needs only 1/33 of the parameters of VGG-16 to achieve the same classification accuracy in ImageNet-1000 classification tasks One of the reasons why we selected this architecture is because it provides a good combination of speed and accuracy when compared to methods such as YOLO[2].Particularly, this

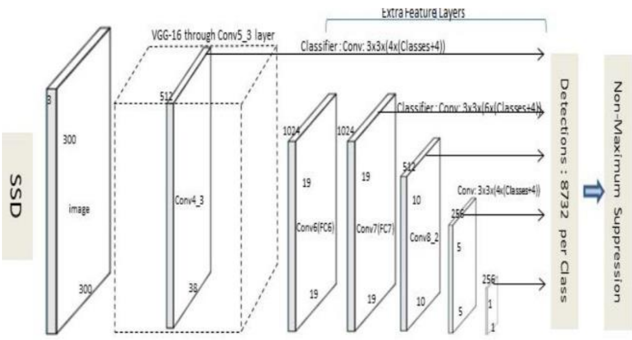

Fig 3 Cropping.

is useful when trying to detect objects in real-time using low power computing devices such as in our case .SSD uses a matching phase while training, to match the appropriate anchor box with the bounding boxes of each ground truth object within an image. Essentially, the anchor box with the highest degree of overlap with an object is responsible for predicting that object's class and its location. This property is used for training the network and for predicting the detected objects and their locations once the network has been trained.

Before training, the data-set of 1233 images was split into training and test subsets in an 80:20 ratio respectively.The model was trained until the loss value stabilized at about .04.The learning rate was set to .001 and the batch size was set to 6.The training was done using Tensorflow-GPU framework.The training lasted for 14000 steps on a NVIDIA MX-150 GPU and an inference graph containing the weights and network nodes were generatedprovides a good combination of speed and accuracy when compared to methods such as YOLO [2]. Particularly, this is true when trying to detect objects in real time in low power computing devices such as

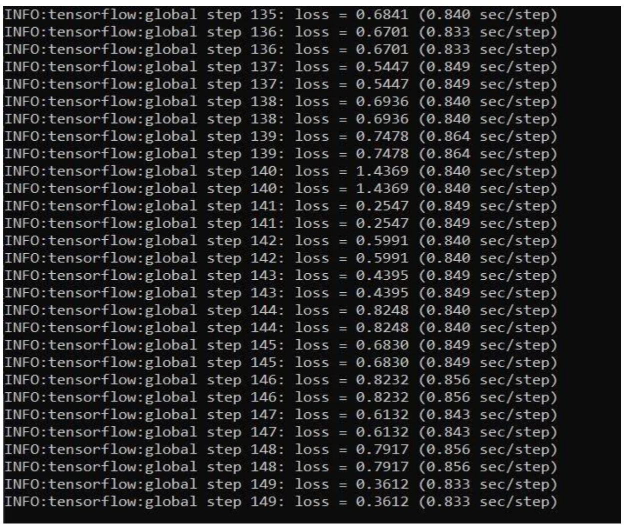
```
INFO:tensorflow:global step 135: loss = 0.6841 (0.840 sec/step)
INFO:tensorflow:global step 136: loss = 0.6701 (0.833 sec/step)
INFO:tensorflow:global step 136: loss = 0.6701 (0.833 sec/step)
INFO:tensorflow:global step 137: loss = 0.5447 (0.849 sec/step)
INFO:tensorflow:global step 137: loss = 0.5447 (0.849 sec/step)
INFO:tensorflow:global step 138: loss = 0.6936 (0.840 sec/step)
INFO:tensorflow:global step 138: loss = 0.6936 (0.840 sec/step)
INFO:tensorflow:global step 139: loss = 0.7478 (0.864 sec/step)
INFO:tensorflow:global step 139: loss = 0.7478 (0.864 sec/step)
INFO:tensorflow:global step 140: loss = 1.4369 (0.840 sec/step)
INFO:tensorflow:global step 140: loss = 1.4369 (0.840 sec/step)
INFO:tensorflow:global step 141: loss = 0.2547 (0.849 sec/step)
INFO:tensorflow:global step 141: loss = 0.2547 (0.849 sec/step)
INFO:tensorflow:global step 142: loss = 0.5991 (0.840 sec/step)
INFO:tensorflow:global step 142: loss = 0.5991 (0.840 sec/step)
INFO:tensorflow:global step 143: loss = 0.4395 (0.849 sec/step)
INFO:tensorflow:global step 143: loss = 0.4395 (0.849 sec/step)
INFO:tensorflow:global step 144: loss = 0.8248 (0.840 sec/step)
INFO:tensorflow:global step 144: loss = 0.8248 (0.840 sec/step)
INFO:tensorflow:global step 145: loss = 0.6830 (0.849 sec/step)
INFO:tensorflow:global step 145: loss = 0.6830 (0.849 sec/step)
INFO:tensorflow:global step 146: loss = 0.8232 (0.856 sec/step)
INFO:tensorflow:global step 146: loss = 0.8232 (0.856 sec/step)
INFO:tensorflow:global step 147: loss = 0.6132 (0.843 sec/step)
INFO:tensorflow:global step 147: loss = 0.6132 (0.843 sec/step)
INFO:tensorflow:global step 148: loss = 0.7917 (0.856 sec/step)
INFO:tensorflow:global step 148: loss = 0.7917 (0.856 sec/step)
INFO:tensorflow:global step 149: loss = 0.3612 (0.833 sec/step)
INFO:tensorflow:global step 149: loss = 0.3612 (0.833 sec/step)
```

Figure 4 training.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

To test the working of the model, it was given as input various images belonging to each class of footwear under varying lighting conditions. Once the footwear was identified by the model, it generates a bar-code corresponding to the identified object detailing its article number and possibly price details.

In this research we used the widely used mean average precision (mAP) to evaluate the performance of our model. The average precision can be defined as the area under the precision recall curve. Our proposed model was able to achieve a 92.1% mAP on our testing subset of 248 images.

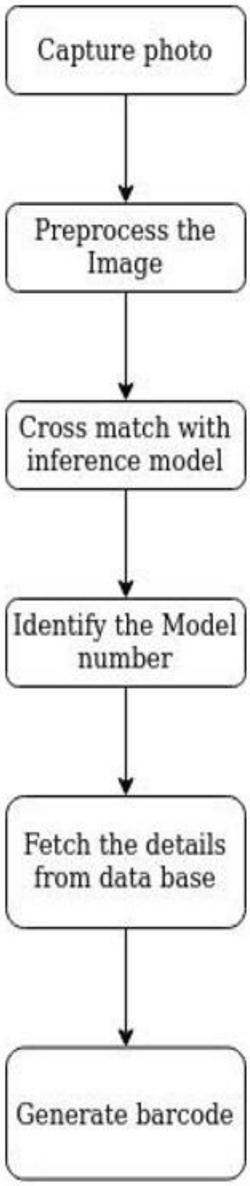

Figure 5 Classification.

International Journal of Scientific Research & Engineering Trends
Volume 6, Issue 4, July-Aug-2020, ISSN (Online): 2395-566X

Figure 6 Classification.

From Fig: 5, we can see that the model correctly identified the article number of the footwear and printed its bar-code. Finally, even though the obtained results are adequate, the accuracy could be improved using a bigger and more complex model. However, in our application, we want to run the detection in real time on a low power device such as the Raspberry Pi single-board computer. That is why despite having lower accuracy, we used a smaller size model in order to obtain a higher detection speed. The proposed model was able to detect shoes at 10 frames per second on an Acer aspire with 2GB Nvidia MX 150 gpu.

V. CONCLUSION

In this research, we proposed a deep learning model to detect in real time the position of the shoes in images. The system could detect the shoes with an average precision similar to other state of the art systems. One limitation of our research is that the architecture of our model was not modified. In future, a modified version of the current model could be developed to increase further the accuracy of our system. The speed of inference can be increased by running the model on an accelerator card such as Nvidia TX1.Running the model using the optimized TensoRT compiler would yield better results. Optimizing the image processing pipeline would also yield better results along with the use of a much bigger data-set. Also, the size of the footwear may be measured using an array of infrared sensors along with the inference model.

REFERENCES

[1] Enzeng Dong, Yao Lu, Shengzhi Du, “An Improved SSD Algorithm and its Mobile Terminal Implementation”-2019 International Conference on Mechatronics and Automation.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016.

[3] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes challenge: A retrospective,” Int. J. Comput. Vis., vol. 111, no. 1, pp. 98–136, 2018

[4] Roland Sczabo, AurelGontean, “Industrial robotic automation with Raspberry PI using image processing”,2018 International conference on applied electronics

[5] Yong Li, Jinbing Xu, “Electronic Product Surface Defect Detection Based on a MSSD Network”, -2020 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)

[6] A. Pouramini and H. Varaee, "A machine vision system for defect detection of a traveling grate conveyor," 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, 2015, pp. 1063-1066, doi: 10.1109/KBEI.2015.74361

[7] J. Jia, "A Machine Vision Application for Industrial Assembly Inspection," 2009 Second International Conference on Machine Vision, Dubai, 2009, pp. 172-176, doi: 10.1109/ICMV.2009.51.

[8] Li, Yiting& Huang, Haisong&Xie, Qingsheng& Yao, Liguo& Chen, Qipeng. (2018). Research on a Surface Defect Detection Algorithm Based on MobileNet-SSD. Applied Sciences. 8. 1678. 10.3390/app8091678.