IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DEFECT DETECTION IN FABRIC USING DEEP LEARNING

[1]Preshika Goel, [2]Geetanjali Gandhi

[1]Student of M.Tech, [2]Head of Department

[1]Computer Science & Engineering Department

[1]Rao Pahlad Singh College of Engineering & Technology, Mahendergarh, Haryana, India

Abstract

A prevalent issue that affects many manufacturing industries and causes significant monetary loss as well as customer loss is the distribution of items to users that have defects or anomalies. These problems are typical in industries that produce components made of fabric, metal, wood, glass, plastic, and paper. These defects are numerous surface defects that vary in size, shape, and color. The best strategies for this task now are deep learning techniques. To locate faults on an object's surface, a vision-based model can be created and trained using standardized defects. It would improve production efficiency and eventually result in added costs for the manufacturing company by making it easier to identify defective items and separate them before selling to the consumer. Deep learning has recently received a lot of attention as a potential tool for automating defect detection.

Keywords : Deep Learning, Defect Detection, Machine Learning, Convolutional Neural Networks.

I. INTRODUCTION

One of the most important tasks in industrial processes for ensuring the proper quality of the finished product is surface inspection. The visual quality impression of a product has a strong influence on whether or not it is purchased. The purpose of this paper is to investigate the use of visual analytics solutions to track surface quality defects or anomalies of products on the industrial shop floor.

The generalized framework can handle defects or anomalies of various shapes, sizes, object colors, and surface defects. Depending on the application, the object with defects or anomalies may be stationary or moving.

In these cases, it is common practice to manually or automatically inspect each fabricated part for visible defects. Manual inspection is a tedious task that frequently results in overlooked errors and subjective assessments. For these and other reasons, the industry has lofty goals for automating any type of surface inspection.

Convolutional Neural Networks (CNNs) are now the preferred method for many image-related machine learning tasks. CNN is one of the fundamental deep learning architectures that can be used to solve visual analytics problems. We can use it for analysis based on the number of layers we chose for our architecture, and by changing some parameters, we can find better hyperparameters for a specific architecture. Other techniques, such as domain adaptation and transfer learning, can provide better results on a given problem statement, where we are attempting to generalize a given knowledge to any of the problem statements based on classes, similarity in dataset instances, and so on.

II. LITERATURE REVIEW

Soon after the release of AlexNet, deep-learning techniques started to be used more frequently for surface-defect categorization problems by Krizhevsky et al. [1]. The work by Masci et al. showed that the deep-learning strategy can perform better for surface-defect classification than traditional machine-vision methods that mix hand-engineered features with support vector machines. [2] In ma- chine vision defect detection, Boaretto and Centeno [3] presented a double image exposure technique for the automatic identification and classification of mechanical product radiation images. The test data of the created classifier reached 88.6 percent accuracy with the usage of semi-supervised learning technology. The discontinuities of “defects” and “no defects” were taken as the indicators for the experiment. One-class SVMs, for example, are learning-based anomaly detection techniques that can only be trained using one class of data which were first introduced by Schö̈̈lkopf et al. [4]. Li et al. [5] used them in combination with clustering for outlier detection of face images. Zhang et al. proposed another one-class anomaly detection technique involving the use of a CNN to map typical examples into a particular feature space, where the mapped instances were then clustered into a hyper sphere [6]. Strongly different photos could be detected using either of these methods, while weakly different images were frequently misclassified. They appear to be less effective for the detection of minute anomalies on surface images. Additionally, there is no pixel-wise anomaly detection offered.

In addition, deep learning [7] [8] is widely used in product defect detection. To perform automatic aperture detection, Yang et al. suggested using a three-point circle and a convolutional neural network (CNN). Time and labor expenses will be cut using the automatic flaw detection system. Surface damage, surface dirt, and stripped screws were taken into consideration for the defect detection problem by Song et al. [9], who also proposed the screw surface defect detection technology based on CNN and demonstrated how deep learning technology outperformed the conventional template matching technology. In summary, the use of deep learning for defect identification is very important.

The main architectures which we have focused upon in the existing work includes, VGG, Resnet, Mobile net, etc. Let’s see the results of all those architectures with some changes in parameters, size of the model, and the top-5 accuracy for model performance.

1. **VGG and its fine-tuned model:** We can achieve an accuracy of 0.901 when using the VGG-16 architecture, which has a size of 528 MB and 138,357,544 parameters. However, with the VGG-19 architecture, which has a size of 549 MB and a number of parameters of 143,667,240, we can achieve an accuracy of up to 0.900.

2. **Resnet and its fine-tuned model:** In terms of Resnet models, Resnet-50 architecture with architecture size of 98 MB, number of parameters 25,636,712, we can get accuracy up to 0.921, but Resnet-101 architecture with architecture size of 171 MB, number of parameters 44,707,176, we can get accuracy up to 0.928, and finally Resnet-152 architecture with architecture size of 232 MB, number of parameters 60,419,944, we can get accuracy up to 0.931. The main takeaway from these results is that increasing the number of layers in the network generally increases accuracy as the feature selection or extraction process is carried out properly.

3. **Mobile-net and its fine-tuned model:** Finally, when it comes to Mobile-net models, simple mobile net architecture with architecture size of 16 MB, number of parameters 4,253,864, accuracy up to 0.895, and mobile net version-2 architecture with architecture size of 14 MB, number of parameters 3,538,984, accuracy up to 0.901, etc.

III. PROPOSED WORK

In our work, we try to classify whether or not there is a defect in the image, and we also try to localize that defect in that image. We used the following data processing techniques to make our data more suitable for feeding into machine or deep learning algorithms:

Image resizing: Because ML and DL algorithms assume that the image we are trying to feed is of the same size or dimension, so that they can work for all those instances uniformly, and as a result there is no bias to any of those instances, we have resized our image to 224*224 to ensure that no important information or features from that image are lost.

Conversion of all multi-dimensional arrays into Dataframe: Because we know that layman arrays take longer to process than numpy arrays, we have converted all of our arrays to numpy format and then converted those arrays into a dataframe, so that processing takes less time.

Add label column in the dataframe: As we are dealing with supervised kind of problem statement, so we gave the output column also in our dataset, so that model can do its retraining with the help of that column and tries to increase the performance on the testing dataset.

Data Augmentation: We observed that defect images are fewer in number than non-defect images in the given dataset, so we used data augmentation to produce nearly the same number of images for both categories. We used two different methods of augmentation, namely positional and color augmentation.

Algorithmic Selections for Defect Detection: Now, when it comes to the classification part, we used the following classification algorithm to determine whether an image contains a defect or not:

1. Simple CNN network
2. Mobile net (with their fine-tuned model)
3. VGG (with their fine-tuning)

Aside from that, we must complete the localization task, which entails creating a bounding box if the image contains defects. We used various versions of the YOLO algorithm, which stands for YOU LOOK ONLY ONCE, such as YOLOv5s, YOLOv5x, YOLOv3, and so on.

We used the following metrics to evaluate the model's performance:

Precision: Precision indicates the performance of a machine learning model that is the quality of a positive prediction made by the model. Precision is described as the ratio of correctly classified positive samples (True Positive) to a total number of classified positive samples (either correctly or incorrectly).

Precision = True Positive / True Positive + False Positive

Recall: The recall is calculated as the ratio between the numbers of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the ability of model to detect positive samples. If the recall is higher, the positive samples detected are more.

Recall = True Positive / True Positive + False Negative

F1-Score: The F1 score has two building blocks which are precision and recall. F1 score's purpose is to combine the precision and recall metrics into a single metric. The F1 score has been designed to work well on imbalanced data at the same time. The F1 score is given as the harmonic mean of precision and recall.

*F1-Score = (2*Precision*Recall / Precision + Recall)*

Accuracy: Accuracy is a metric which describes how the model performs across all classes. When all classes are of equal importance accuracy is useful. Accuracy is defined as the ratio between the numbers of correct predictions to the total number of predictions.

Accuracy=True Positive + True Negative / True Positive + True Negative + False Positive + False Negative

IV. EXPERIMENTATIONS AND RESULTS

4.1 Classification Experimentations

Simple CNN: Three Conv2D/MaxPooling2D pairs serve as the feature extractor, and three Dense Layers serve as the classifier. Validation accuracy can reach up to 52.78 percent.

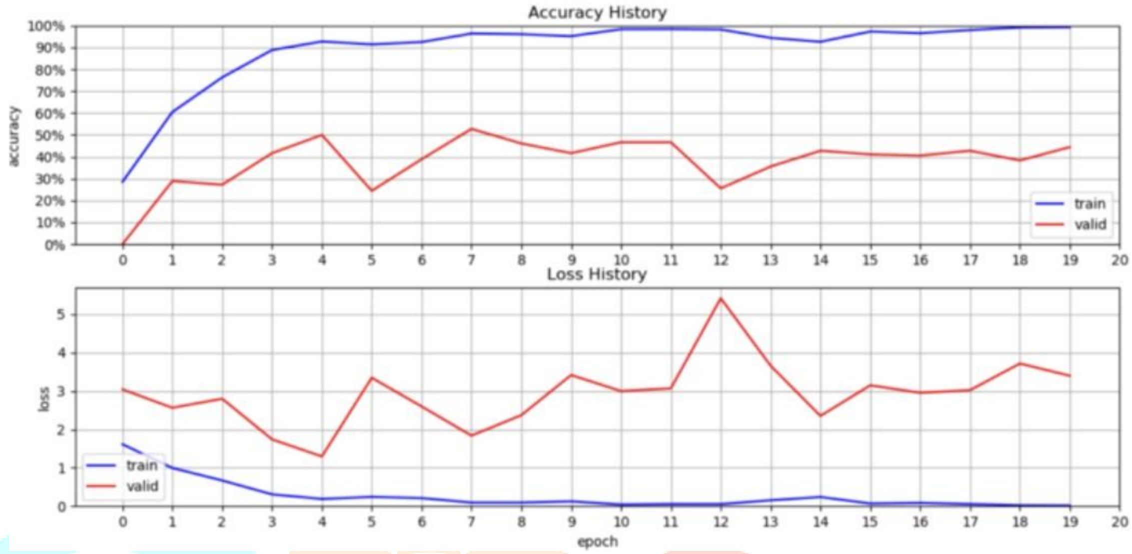

Figure 1: Loss and Accuracy vs. Epochs Curve for Simple CNN

Mobile net with fine-tuning: Validation accuracy can reach 91.03 percent. On epoch 13, this model achieves 89.6 percent, then 91.03 percent on epoch 50.

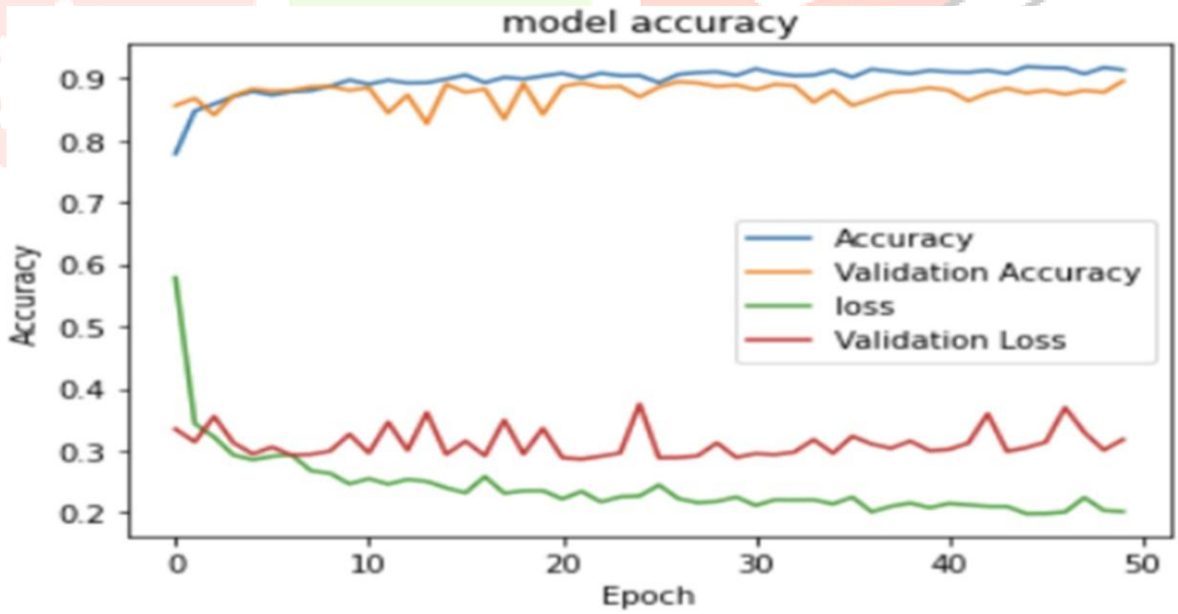

Figure 2: Loss and Accuracy vs. Epochs Curve for Mobile net

VGG-16 with fine-tuning: Validation accuracy can reach 87.79 percent. On epoch 100, this model achieves 84.6 percent, then 87.79 percent on epoch 250.

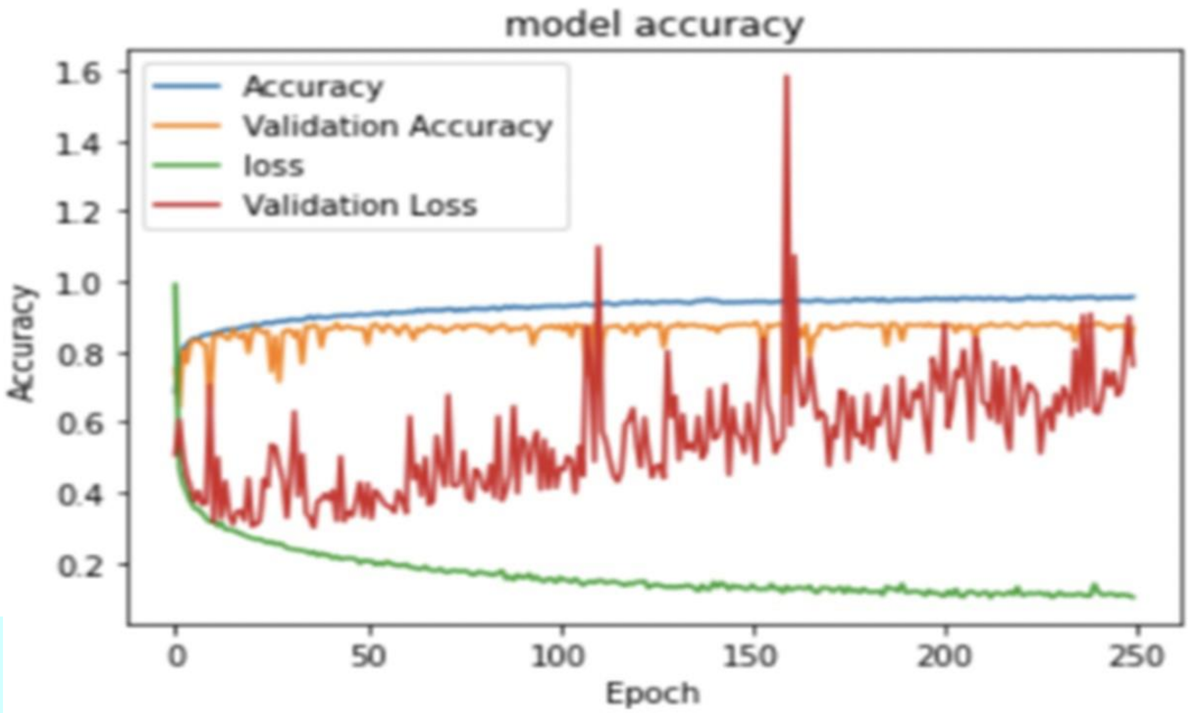

Figure 3: Loss and Accuracy vs. Epochs Curve for VGG

4.2. Localization Experimentations

4.2.1. YOLOv3: Because the dataset provided to us was ideal for YOLO implementation, YOLO v3 object detection was used to detect flaws in the dataset images.

4.2.2. YOLOv5x: Following that, YOLOv5x was trained to locate the flaws. YOLOv5x outperformed YOLOv3 in both accuracy and speed. In 18 hours, the model was trained.

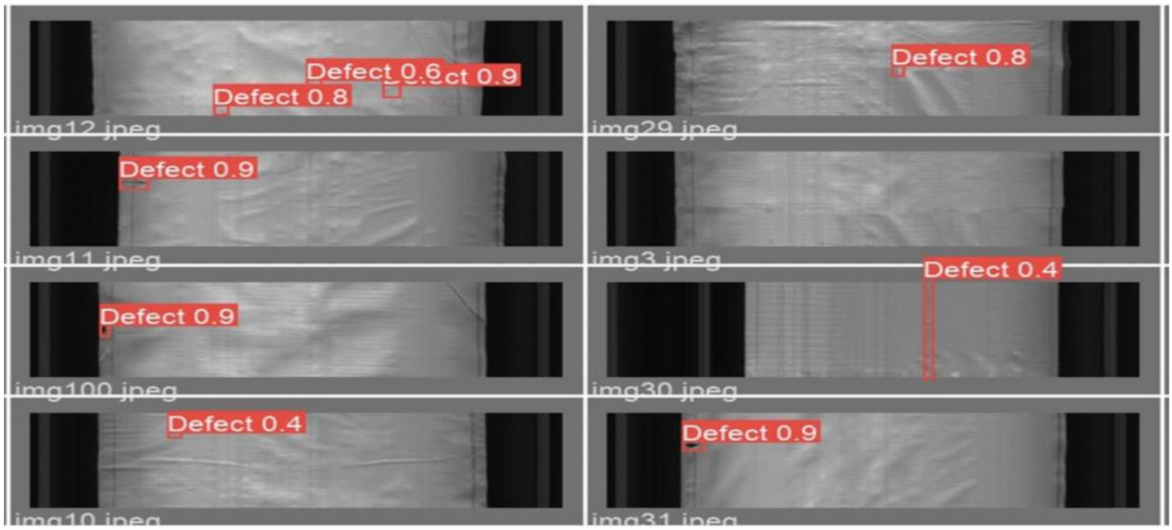

Figure 4: Defect Detection on some set of images by YOLOv5x

4.2.3. YOLOv5s: Yolov5s was programmed to produce the desired results. The model was trained for 500 epochs with a batch size of 32.On the first attempt, a precision of 85 percent was achieved, with a recall of 70 percent.

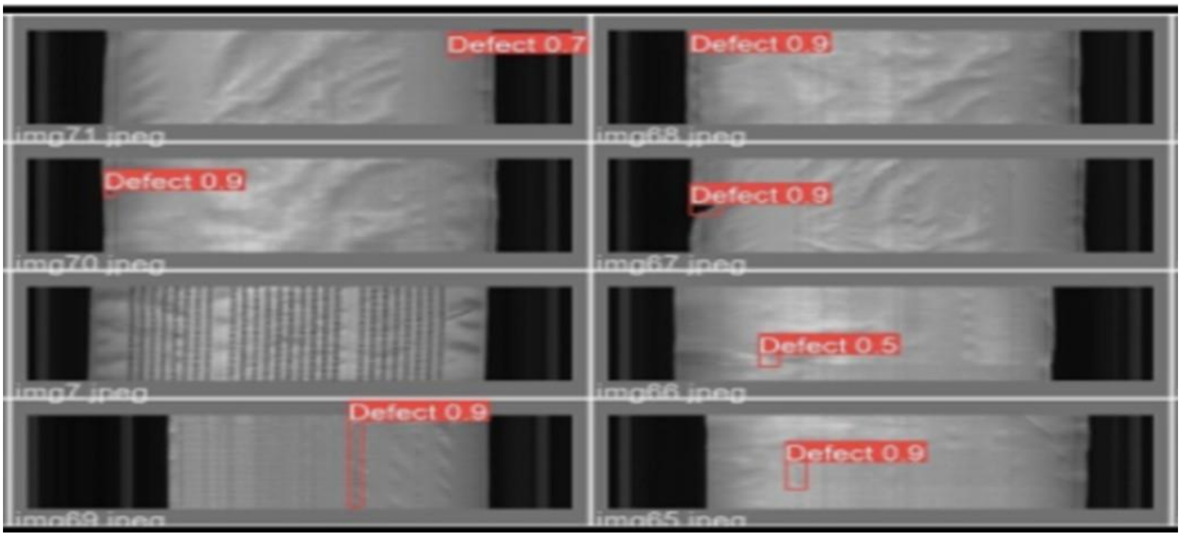

Figure 5: Defect Detection on some set of images by YOLOv5s

V. CONCLUSION

5.1 Classification:

Based on the metrics chosen, VGG-16 outperforms all other algorithms for classification tasks because it is built as a deep CNN and outperforms baselines on many tasks and datasets other than ImageNet. As a result, VGG remains one of the most widely used image-recognition architectures. Finally, this architecture is an innovative object-recognition model that supports up to 19 layers, but 16 layers are sufficient for our problem statement.

5.2 Localization:

Based on the metrics chosen, YOLOv5s outperforms all other algorithms for localization tasks. The following are the primary reasons why YOLOv5s outperforms other versions of the YOLO algorithm:

1. It is approximately 88 percent smaller than YOLOv4 (27 MB vs. 244 MB)
2. It is approximately 180 percent faster than YOLOv4 (140 FPS vs. 50 FPS)
3. On the same task, it is roughly as accurate as YOLOv4 (0.895 mAP vs. 0.892 mAP)

VI. FUTURE WORK

Filtering using any supervised machine learning algorithm:

As we have seen, when we use an undefected image in a model, we have to pass that image through the entire architecture, which takes longer than the optimal time, so to overcome this problem, we can use some filtering algorithm, such as supervised training, which helps us to initially filter those defected and undefected images, and as a result, for those undefected images, we can simply skip those architecture (deep learning models), and as a result, latency of the module reduces. To do the same, we can primarily use KNN type algorithms, which have fewer hyperparameters and also overcome the problem of heavier computation.

VII. REFERENCES

[1] Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet Classification with Deep Convolutional Neural Networks. In: Advances In Neural Information Processing Systems 25, pp 1097–1105.

[2] Masci J, Meier U, Ciresan D, Schmidhuber J, Fricout G (2012) Steel defect classification with Max-Pooling Convolutional Neural Networks. In: Proceedings of the International Joint Conference on Neural Networks, DOI 10.1109/IJCNN.2012.6252468.

[3]N. Boaretto and T. M. Centeno, "Automated detection of welding defects in pipelines from radiographic images DWDI," NDT E Int., vol. 86, pp. 7-13, Mar. 2017.

[4]B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional," Neural computation, vol. 13, no. 7, pp. 1443–1471, 2001.

[5]S. Li, M. Shao, and Y. Fu, "Locality linear fitting one-class svm with low-rank constraints for outlier detection: 6 - 11 July 2014, beijing, china; [part of the 2014 ieee world congress on computational intelligence (ieee wcci 2014)]," pp. 676–683, 2014. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=6880678

[6]M. Zhang, J. Wu, H. Lin, P. Yuan, and Y. Song, "The application of one-class classifier based on cnn in image defect detection," Procedia Computer Science, vol. 114, pp. 341–348, 2017.

[7]Y. Yang, Y. Lou, M. Gao, and G. Ma, "An automatic aperture detection system for LED cup based on machine vision," Multimedia Tools Appl., vol. 77, no. 18, pp. 23227-23244,Sep. 2018.

[8]J. C. P. Cheng and M. Wang, "Automated detection of sewer pipe defects in closed-circuit television images using deep learning techniques," Automat. Construct., vol. 95, pp. 155-171, Nov. 2018.

[9]L. Song, X. Li, Y. Yang, X. Zhu, Q. Guo, and H. Yang, "Detection of micro-defects on metal screw surfaces based on deep convolutional neural networks," Sensors, vol. 18, no. 11, pp. 3709-3723, Oct. 2018.