



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 Issue: IV Month of publication: April 2022

DOI: <https://doi.org/10.22214/ijraset.2022.41978>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Metal Surface Defect Detection using Segmentation and Decision Networks

Pranav Dengale¹, Raj Karkhanis², Bhargav Phadke³, Prof. Vaishali Gaidhane⁴

^{1, 2, 3, 4}Department of Computer Engineering, Datta Meghe College of Engineering, Navi Mumbai

Abstract: *With set Quality Control and Quality Assurance processes, defects in manufacturing are rare but when they occur they can result in huge losses for the manufacturer if unchecked. Detecting these defects with the help of machine learning and deep learning techniques is now an interesting and promising area of research and industrial application. The project aims to build a model that will detect surface level cracks on metal commutators with deep learning techniques like image segmentation. Since in industry, defects are rare the segmentation based deep learning model will learn on fewer training samples as compared to a typical deep learning model which would require hundreds and thousands of training samples. The model is trained on the publicly available dataset provided by the Kolektor Group, which contains images of commutators and annotated if there are any surface cracks on the commutators.*

Keywords: *Quality Assurance, Defect detection, commutators defect detection, Kolektor Dataset*

I. INTRODUCTION

Metal plates are indispensable materials for the car business, national defence business, machinery producing, industry, light-weight business, etc. In industrial processes, one of the foremost necessary tasks once it involves guaranteeing the proper quality of the finished product is review of the product's surfaces. However, thanks to the issues of raw materials and technology, varied varieties of defects are going to be made within the production method of metal plates particularly cracks, scabs, curling edges, cavities, abrasions, and different defects on the surface. These have a fatal impact on the corrosion resistance and strength of the plate, and affect the economic benefits of the mill. At present, the surface defect detection of strip steel largely adopts the maneuver of manual detection. This methodology is unquestionably affected by the subjective factors of the testing personnel. Such management is, however, terribly time overwhelming, inefficient, and might contribute to a heavy limitation of the assembly capability. Moreover, the accuracy of the check results is low, and also the responsibility is poor. Therefore, it's essential to review the formula of automatic period of time detection of surface defects on the production line.

In the history, classic machine-vision styles were sufficient to address these issues; still, with the Industry 4.0 paradigm the trend is moving towards the generalisation of the product line, where rapid-fire adaption to a new product is needed. Classical machine-vision styles are unfit to insure similar inflexibility. Hand-engineering of features, thus, plays an important part in classical approaches, but similar features aren't suited for different tasks and lead to long development cycles when machine-vision styles must be manually acclimated to different products. A result that allows for bettered inflexibility can be plant in data-driven, machine-learning approaches where the developed styles can be snappily acclimated to new types of products and face blights using only the applicable number of training images. This paper focuses on using state-of-the-art machine literacy styles to address the discovery of visual face blights. The focus is primarily on deep literacy styles that have, in recent times, come the most common approach in the field of computer vision. When applied to the problem of face quality control, deep literacy styles can achieve excellent results and can be acclimated to different products. Compared to classical machine vision styles, deep literacy can directly learn features from low position data, and has a advanced capacity to represent complex structures, therefore fully replacing hand engineering of features with automated literacy processes. With a rapid-fire adaption to new products this system becomes veritably suitable for the flexible product lines needed in Industry 4.0. Segmentation are going to be done by a Convolution Neural Network (CNN), using eleven convolution layers and three max-pooling layers that reduce the resolution by an element of two. Each convolution layer is followed by a feature normalisation and a non-linear ReLU layer, which helps in increasing the speed of convergence during the training. Feature normalisation normalises each channel to a zero mean and unit variance distribution. First nine convolution layers use a 5x5 kernel, the last two use a 15x15 and a 1x1 kernel respectively. A final output mask is obtained after the last convolution layer. The design of the segmentation stage focuses on detecting small defects on the surface in large-scale resolution images. The network is meant for 2 things:

A large receptor size: a further down sampling layer and enormous kernel sizes are wont to increase the receptor size.

Ability to capture small feature details: Number of layers between down-sampling layers are arranged such more layers are within the higher sections and fewer layers within the lower sections, this is often increases capacity of features with high receptor size. Finally, the down-sampling is achieved using max-pooling rather than convolutions with an outsized stride. This ensures small but important details survive the down-sampling process, which is especially important during this network with additional down-sampling layers.

II. RELATED WORK

Automated surface anomaly detection is an exciting and promising area of research as well as industrial applications. With advancements in machine learning and deep learning techniques in the domain of visual application, detecting anomalies or irregularities has become a reality. It is difficult to properly identify and control these defects thanks to the various inspection techniques on different materials and sizes. The early detection of faults or defects and therefore the removal of the weather which will produce them are essential to enhance product quality and reduce the economic impact caused by discarding defective products. Otsu method is a very common technique for thresholding of automatic machine vision inspection of defects. Otsu method works very well with bimodal distributions but fails for unimodal or close to unimodal distributions. Since defects can range from no to small to large defects, the grey level distribution will range from unimodal to bimodal. (Ng, 2006) discusses this revised Otsu method as the valley-emphasis model for defect detection.

Some metals such as aluminium, gold, copper, and silver reflect more light than other metals, (Zhang, 2011) designed a vision system for detecting surface-level defects for these metals. The study uses the wavelet smoothing method to eliminate noise from images, which are then segmented with the Otsu threshold. Deep learning methods have recently been used to address the problem of surface defect detection in industrial quality control. Training requires a large amount of data and often requires high-precision labels, which makes it difficult to solve many industrial problems or significantly increases the cost of the solution due to annotation requirements. This task completely eases the high demands of supervised learning methods and reduces the need for highly detailed annotations. By proposing a deep learning architecture, we will investigate the use of annotations in a variety of details for surface defect detection tasks, from faint (image-level) labels to mixed monitoring to full (pixel level) annotations on the task of surface-defect detection. With deep learning like Convolution Neural Networks (CNNs), (Soukup & Huber-Mörk, 2014) apply CNNs to detecting defects on a database of photometric stereo images of metal surface defects, i.e. rail defects. The defects are illuminated by differently-coloured lights from different and constant directions and are captured in a photometric light-dark field setup. This paper experiments with classical CNNs with purely supervised learning for defect detection.

(Tao, 2018) explores a novel cascaded autoencode architecture (CASAE) for segmenting and localising defects. The paper follows a twofold procedure that localises and classifies defects. The cascading network transforms images into a pixel-level prediction mask and is then classified into specific classes via a compact CNN.

(Yun, 2020) proposes a new convolutional variational autoencoder (CVAE) and deep CNNs to detect defects in manufacturing. Images are generated with a conditional CVAE (CCVAE) for each defect type in a single CVAE model. A deep CNN classifier using data generated from CCVAE is used to verify the performance of the proposed model.

(Bulnes, FG, 2016) proposes a method for detecting certain types of defects that can occur during the manufacture of web materials: periodic defects. This type of defect is extremely harmful as it creates many surface defects, significantly reduces the quality of the final product, and in some cases can be unfit for sale. Two different functions must be performed very often in order to carry out the proposed method. It is very important to use as little time as possible, as the time available to detect these defects may be limited. An analysis is performed on how the method accesses the input data to reduce the overall time required for recognition. Therefore, the most efficient data structure for storing information is determined. (Grasso, 2017) reviews the literature and therefore the commercial tools for in situ monitoring of powder bed fusion (PBF) processes. It explores the various categories of defects and their main causes, the foremost relevant process signatures and therefore the in situ sensing approaches proposed up to now. Particular attention is dedicated to the development of automated defect detection rules and therefore the study of process control strategies, which represent two critical fields for the development of future smart PBF systems. (JM Fraser, 2016) A low coherence interferometric imaging technique, termed inline coherent imaging (ICI), is coaxially integrated into the selective laser melting process to monitor melt pool morphology changes and stability at 200 kHz. Galvanometerbased scanning of the imaging spot relative to the melt pool allowed the capture of the geometrical profile of the melt pool and environment undetectable by existing thermal-based imaging systems. Exploiting the highspeed imaging rate, time-resolved ICI measurements demonstrate that melt pool fluctuations strongly influence final track quality. the rate of convergence during the learning. Feature normalisation normalises each channel to a zero mean and unit variance distribution. First nine convolution layers use a 5x5 kernel, the last two use a 15x15 and a 1x1 kernel respectively. A final output mask is obtained after the last convolution layer

The design of the segmentation stage focuses on detecting small defects on the surface in large-scale resolution images. The network is designed for two things:

- 1) A large receptor size: An additional down sampling layer and large kernel sizes are used to increase the receptor size.
- 2) Ability to capture small feature details: Number of layers between down-sampling layers are arranged such that more layers are in the higher sections and fewer layers in the lower sections, this increases capacity of features with high receptor size

Finally, the down-sampling is achieved using max-pooling instead of convolutions with a large stride. This ensures small but important details survive the down-sampling process, which is particularly important in this network with additional down-sampling layers.

A. Decision Network

The next stage, the decision stage, takes the output of the segmentation stage as input and performs binary classification and decides whether the given image has a defect or not. This is a supervised binary classification task resulting in either the image has a defect or doesn't have a defect.

The network takes output of the last convolution layer, 1024 layers, concatenated with a single-channel segmentation output map. This results in a 1025 channel volume that represents the input for the remaining layers with a max-pooling layer and a convolutional layer with 5x5 kernel sizes. The number of layers was chosen to increase as the resolution of the image decreases so that the convolution requirement for each layer stays the same.

The design of the decision network follows two principles:

- 1) Several layers of convolution and down-sampling is used to ensure capture of not only local shapes but also global ones that span a large area of the image
- 2) Decision network uses a final segmentation output map that allows decision net to avoid large segmentation maps and also reduces over-fitting

Process errors due to improper parameter regimes are detected and characteristic error signatures are identified.

(GS Hong, 2018) proposes a new method for error detection in SLM parts. Setup was flexible using a microphone and error detection was achieved via the Deep Belief Network (DBN) framework. It is implemented by a simplified classification structure with no signal preprocessing or feature extraction. Experimental results have shown that it is feasible to use acoustic signals for quality monitoring and that the DBN approach can achieve high failure detection rates under five melt conditions without signal pretreatment.

III. ARCHITECTURE

The problem of detecting surface level defects in metals is a binary classification task. The goal is to accurately classify an image of a metal having defects as anomaly rather than locating the defect on the metal surface. To combat the problem of a small set of training samples, the proposed approach is a two-stage design. The first stage is the segmentation stage, followed by the next stage of decision stage.

A. Segmentation Network

Segmentation will be done by a Convolution Neural Network (CNN), using eleven convolution layers and three max-pooling layers that reduce the resolution by a factor of two. Each convolution layer is followed by a feature normalisation and a non-linear ReLU layer, which helps in increasing

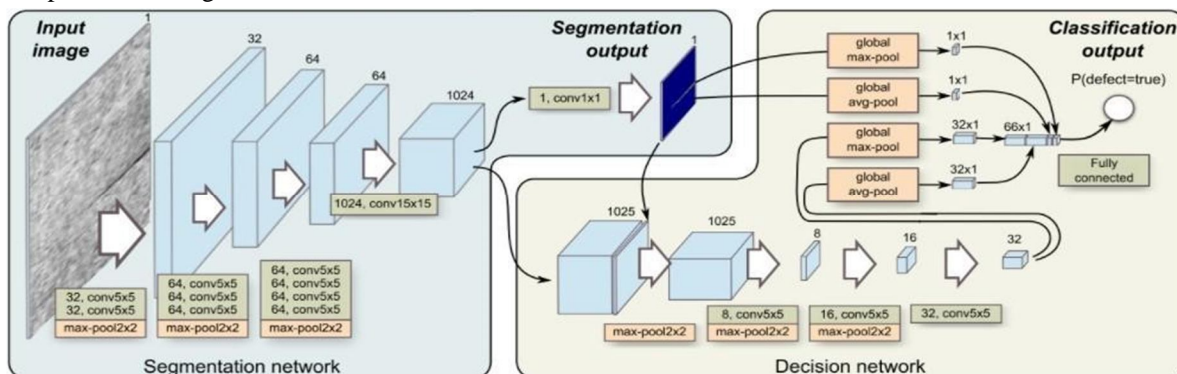


Figure 1: Block design of Segmentation and Design Stage

B. Incremental Model

Incremental Model is a process of software development where requirements are broken down into multiple standalone modules of the software development cycle. Incremental development is done in steps from analysis, design, implementation, testing/verification, maintenance.

Each iteration passes through the requirements, design, coding, and testing phases. Each subsequent release of the system adds function to the previous release until all designed functionality is built.

The system is put into production when the first iteration is delivered. The first increment is often a core product where the basic requirements are addressed, and supplementary features are added in the next increments. Once the core product is analysed by the client, the next iteration is developed.

C. Google Colab

In our earlier iterations we tried to develop the project on our local machines, but the neural network training was unsuccessful due to hardware limitations.

Later on we tried to implement an AWS based solution that was an application running on an edge machine with a web interface, but that too was only trained halfway.

We then switched to Google Colaboratory. Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

We created a new notebook, mounted our Google Drive, downloaded and extracted the dataset, trained the segmentation network and used those learned weights for training the decision network, and finally tested the entire network and displayed accuracy metrics.

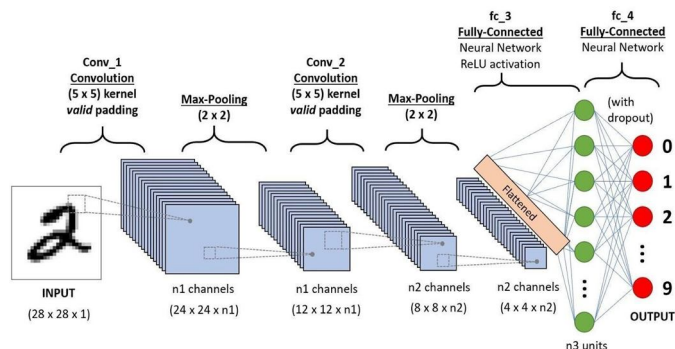


Figure 2: Convolutional Neural Networks

A CNN takes an image as an input and applies importance like learnable weights and biases to various objects of the image and be able to differentiate one from the others. The preprocessing required in a CNN is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

A CNN is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

The role of the CNN is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction. The objective of the Convolution Operation is to extract the high-level features such as edges, from the input image. Segmentation will be done by a Convolution Neural Network (CNN), using eleven convolution layers and three max-pooling layers that reduce the resolution by a factor of two.

This results in a single-channel output map with an 8-times-reduced resolution of the input image. Drop-out is not utilised in this approach, since the weight sharing in convolutional layers provides sufficient regularisation.

The network takes the output of the last convolutional layer of the segmentation network (1024 channels) concatenated with a single-channel segmentation output map. This results in a 1025-channel volume that represents the input for the remaining layers with a max-pooling layer and a convolutional layer with 5x5 kernel sizes.

The input into the proposed network is a grey-scale image. The network architecture is independent of the input size, similar to fully convolutional networks, since fully connected layers are not used in feature maps, but only after the spatial dimension is eliminated with global average and max pooling. Input images can therefore be of a high or a low resolution, depending on the problem. Two image resolutions are explored in this paper: 1408×512 and 704×256 .

D. Image Segmentation

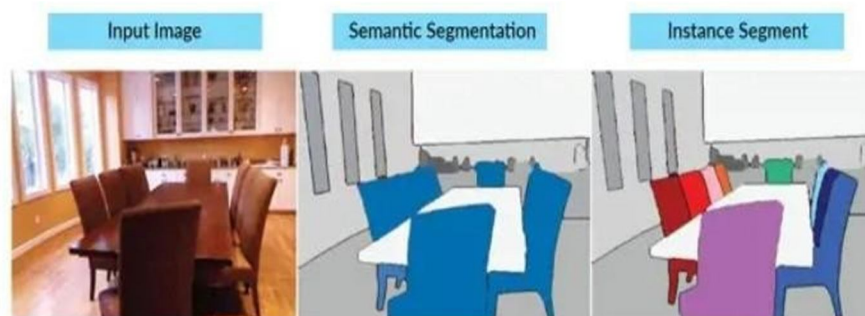


Figure 3: Image Segmentation

Image segmentation is the process by which a digital image is partitioned into various subgroups (of pixels) called Image Objects, which can reduce the complexity of the image, and thus analysing the image becomes simpler.

We use various image segmentation algorithms to split and group a certain set of pixels together from the image. By doing so, we are actually assigning labels to pixels and the pixels with the same label fall under a category where they have some or the other thing common in them.

Using these labels, we can specify boundaries, draw lines, and separate the most required objects in an image from the rest of the not-so-important ones. In the below example, from a main image on the left, we try to get the major components, e.g. chair, table etc. and hence all the chairs are colored uniformly. In the next tab, we have detected instances, which talk about individual objects, and hence all the chairs have different colours.

This is how different methods of segmentation of images work in varying degrees of complexity and yield different levels of outputs. In the absence of publicly available datasets with real images of annotated surface defects a replacement dataset termed Kolektor Surface Defect Dataset was created.

The dataset is made from images of defective electrical commutators that were provided and annotated by Kolektor Group d. o. o.. Specifically, microscopic fractions or cracks were observed on the surface of the plastic embedding in electrical commutators. The area of every commutator was captured in eight non-overlapping images.

The images were captured during a controlled environment, ensuring high-quality images with a resolution of 1408×512 pixels. The dataset consists of fifty defected electrical commutators, each with up to eight relevant surfaces. This resulted during a total of 399 images. In two items the defect is visible in two images while for the remaining items the defect is merely visible during a single image, which suggests there have been 52 images where the defects are visible (i.e., defective or positive samples). For every image, an in depth pixel-wise annotation mask is provided. The remaining 347 images function negative examples with non-defective surfaces.

The proposed network is first evaluated under several different training setups, which include differing types of annotations, input-data rotation and different loss functions for the segmentation network. Altogether, the network was evaluated under four configuration groups: five annotation types two loss-function types for the segmentation network (mean squared error and cross-entropy) two sizes of the input image (full size and half size) without and with 90° input-image rotation.

For the purpose of this evaluation, the problem of surface defect detection is translated into a binary image classification problem. The main objective is to classify the image into two classes: (a) defect is present and (b) defect is not present. Although pixel-wise segmentation of the defect can be obtained from the segmentation network the evaluation does not measure the pixel-wise error, since it is not crucial in industrial settings. Instead, only the per-image binary image classification error is measured. The segmentation output is only used for visualisation purposes.

IV. RESULTS

We executed a testing script to test the segmentation and decision network with some test images to evaluate performance. We achieved 97.9% Average Precision, 1 False Positives, and 2 False Negatives.

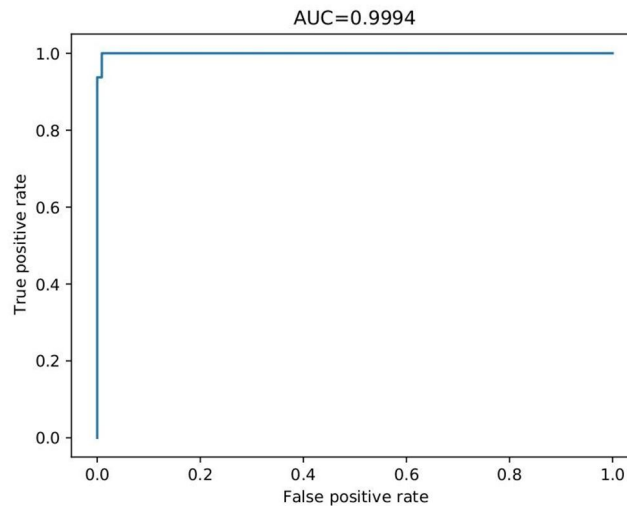


Figure 4: Area Under Curve

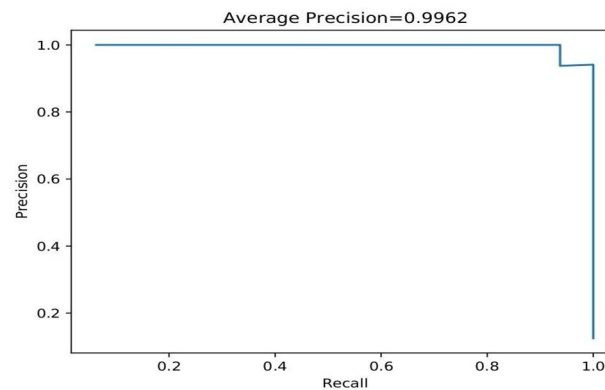


Figure 5: Precision Recall Curve

```

2022-04-27 14:59:42.828424: step 6270, loss = 0.00012 (3.2 examples/sec; 0.317 sec/batch)
2022-04-27 14:59:46.215962: step 6281, loss = 0.00000 (3.3 examples/sec; 0.307 sec/batch)
2022-04-27 14:59:49.734686: step 6292, loss = 0.00000 (2.6 examples/sec; 0.392 sec/batch)
2022-04-27 14:59:53.223678: step 6303, loss = 0.00001 (2.5 examples/sec; 0.392 sec/batch)
2022-04-27 14:59:56.753234: step 6314, loss = 0.00002 (3.1 examples/sec; 0.325 sec/batch)
2022-04-27 15:00:00.178815: step 6325, loss = 0.00000 (3.1 examples/sec; 0.325 sec/batch)
2022-04-27 15:00:03.740839: step 6336, loss = 0.00002 (3.2 examples/sec; 0.316 sec/batch)
2022-04-27 15:00:07.238404: step 6347, loss = 0.00000 (3.0 examples/sec; 0.331 sec/batch)
2022-04-27 15:00:10.648555: step 6358, loss = 0.00000 (3.4 examples/sec; 0.294 sec/batch)
2022-04-27 15:00:14.432851: step 6369, loss = 0.00011 (3.3 examples/sec; 0.301 sec/batch)
2022-04-27 15:00:18.562187: step 6380, loss = 0.00063 (3.4 examples/sec; 0.298 sec/batch)
2022-04-27 15:00:22.936339: step 6391, loss = 0.00022 (3.3 examples/sec; 0.307 sec/batch)
2022-04-27 15:00:27.536369: step 6402, loss = 0.00002 (3.4 examples/sec; 0.291 sec/batch)
2022-04-27 15:00:31.985206: step 6413, loss = 0.00000 (3.4 examples/sec; 0.297 sec/batch)
2022-04-27 15:00:36.312593: step 6424, loss = 0.00001 (3.4 examples/sec; 0.294 sec/batch)
2022-04-27 15:00:40.639510: step 6435, loss = 0.00000 (3.0 examples/sec; 0.332 sec/batch)
2022-04-27 15:00:45.972390: step 6446, loss = 0.00003 (3.4 examples/sec; 0.292 sec/batch)
2022-04-27 15:00:49.467236: step 6457, loss = 0.00000 (3.3 examples/sec; 0.301 sec/batch)
2022-04-27 15:00:53.999554: step 6468, loss = 0.00018 (3.0 examples/sec; 0.331 sec/batch)
2022-04-27 15:00:58.379954: step 6479, loss = 0.00000 (3.4 examples/sec; 0.298 sec/batch)
2022-04-27 15:01:02.870338: step 6490, loss = 0.00009 (3.4 examples/sec; 0.292 sec/batch)
2022-04-27 15:01:07.351926: step 6501, loss = 0.00000 (3.4 examples/sec; 0.293 sec/batch)
2022-04-27 15:01:11.802874: step 6512, loss = 0.00000 (3.3 examples/sec; 0.307 sec/batch)
2022-04-27 15:01:16.273845: step 6523, loss = 0.00007 (3.0 examples/sec; 0.330 sec/batch)
2022-04-27 15:01:20.621739: step 6534, loss = 0.00000 (3.4 examples/sec; 0.297 sec/batch)
2022-04-27 15:01:24.063891: step 6545, loss = 0.00000 (3.4 examples/sec; 0.291 sec/batch)
2022-04-27 15:01:28.559854: step 6556, loss = 0.00000 (3.3 examples/sec; 0.299 sec/batch)
2022-04-27 15:01:33.079903: step 6567, loss = 0.00000 (3.2 examples/sec; 0.308 sec/batch)
2022-04-27 15:01:37.572600: step 6578, loss = 0.00009 (3.2 examples/sec; 0.317 sec/batch)
2022-04-27 15:01:42.066121: step 6589, loss = 0.00000 (3.4 examples/sec; 0.297 sec/batch)

```

Figure 6: Decision Net Fold 0 training

Here are some training output metrics. We trained with a 3 fold validation for both segmentation and decision neural networks. Both networks were trained upto 6600 steps. With 33 defective images per training set in one fold and alternating between defective and non-defective images for equal weight to each.

Along with precision and area curves we also generated segmented outputs of the original image which was used for the decision network.

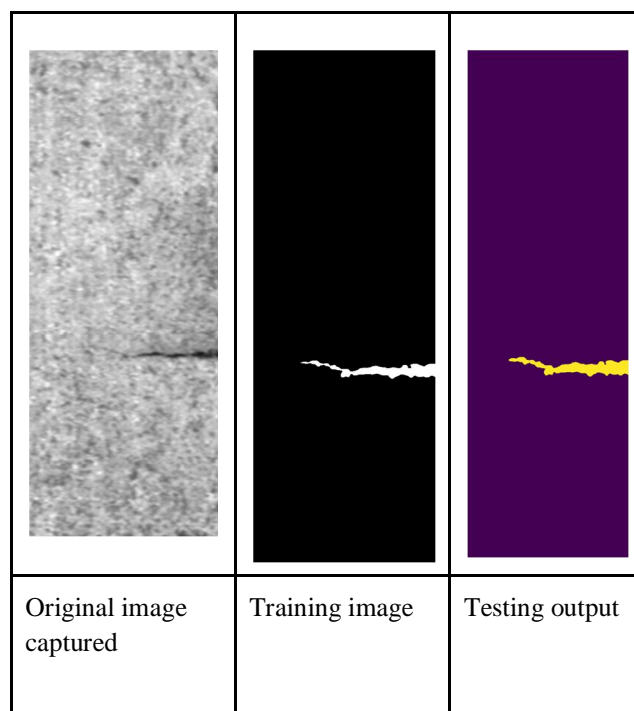


Figure 7: Outputs

V. CONCLUSION

A two-stage approach was presented. The first stage included a segmentation network trained on pixel-wise labels of the defect, while the second stage included an additional decision network built on top of the segmentation network to predict the presence of the anomaly for the whole image. An extensive evaluation of the proposed approach was made on a semi-finished industrial product, i.e., an electrical commutator, where the surface defects appeared as fractures of the material. This problem domain has been made publicly available as a benchmark dataset, termed the Kolektor Surface Defect Dataset.

VI. FUTURE SCOPE

This model is prepared on an after-the-fact dataset, and not on an actual industry line. In the future, it would be exciting to see it in action connected to a live camera that will capture images of the parts and feed it to the model which will do the defect detection in situ. Furthermore, this model was only trained to detect surface-level cracks, in the industry, there are more defects that can pop up, and thus this system isn't holistic.

REFERENCES

- [1] Le, X., Mei, J., Zhang, H., Zhou, B., & Xi, J. (2020). A learning-based approach for surface defect detection using small image datasets. *Neurocomputing*, 408(September 2020), 112-120. <https://doi.org/10.1016/j.neucom.2019.09.107>
- [2] Ng, H.-F. (2006). Automatic thresholding for defect detection. *Pattern Recognition Letters*, 27(14), 1644-1649. <https://www.sciencedirect.com/science/article/pii/S016786550600119X>
- [3] Soukup, D. (2014). *Convolutional Neural Networks for Steel Surface Defect Detection from Photometric Stereo Images*. Springer.
- [4] https://link.springer.com/chapter/10.1007/978-3-319-14249-4_64 Tao, X. (2018). Automatic Metallic Surface Defect Detection and Recognition with Convolutional Neural Networks. *Applied Sciences*, 8(9), 1575. <https://doi.org/10.3390/met11030388>
- [5] Wang, S., XiaojunXia, X., Ye, L., & yang, B. (2021). Automatic Detection and Classification of Steel Surface Defect Using Deep Convolutional Neural Networks. *Metals*, 2021(11(3)), 388. <https://doi.org/10.3390/met11030388>
- [6] Ye, D., Yong, G. S., Zhang, Y., Zhu, K., & Hsi Fuh, J. Y. (2018).
- [7] Defect detection in selective laser melting technology by acoustic signals with deep belief networks. *The International Journal of Advanced Manufacturing Technology*, 96(May 2018), 2791–2801. <https://doi.org/10.1007/s00170-018-1728-0>
- [8] Yun, J. P. (2020). Automated defect inspection system for metal surfaces based on deep learning and data augmentation. *Journal of Manufacturing Systems*, 55(April 2020), 317-324. <https://www.sciencedirect.com/science/article/abs/pii/S027861252030042X>
- [9] Zhang, X. (2011). A vision inspection system for the surface defects of strongly reflected metal based on multi-class SVM. *Expert Systems with Applications*, 38(5), 5930-5939. <https://www.sciencedirect.com/science/article/abs/pii/S0957417410012674>



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)