International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Design and Development of a Deep Learning Model for Electronic Manufacturing Industry Using TeetuNet

Pankaj Kumar Sharma[1], Dr. Sandhya Sharma[2]

Submitted: 14/03/2024 Revised: 29/04/2024 Accepted : 6/05/2024

Abstract*:* One of the biggest challenges in electronic manufacturing industry is to take care of quality check process of manufacturing semiconductor wafers. Post pandemic the supply chain for electronic chips has disturbed badly and the supply is much less than the demand all over the world. The present systems are not as efficient and smart to reduce the testing time of semiconductor wafer drastically. Many a times it so happens that one or the other kind of defects are present in the manufactured wafer which degrades the quality of wafer and adds up more time to produce same number of wafers. Therefore a much more efficient, reliable quality check system is necessary to tackle the issue. Since the advantage of Artificial Intelligence is in almost every field nowadays, in this paper we propose a much more capable Deep Learning model based on CNN, which can detect a defected and non defected wafer using image. Unlike the present systems where all the testing work is done using sensor data points, this Deep Learning model process on the image of wafer and gives results with greater accuracy.

Keywords: *Artificial Intelligence, Semiconductor wafer, Electronic chip Manufacturing, Deep Learning*

1. Introduction

Semiconductor wafers are the backbone of every electronic item used in multiple products today. Be it Healthcare, Education, Communication, E-commerce, Satellite technology, Space science, Automobile industry, Hospitality, Public Administration, Aerospace, Solar technologies, optics, these wafers are extensively used in every field as part of different multipurpose devices and apparatus. Semiconductor wafer manufacturing is a very complex process which comprises of many steps such as Slicing, Lapping, Etching, Polishing, Doping, Cleaning and Inspection.

Wafer inspection and quality check step comes at the last and plays a significant role in deciding the defected wafer. At present many technical procedures are applied to keep this process quicker and efficient but most of the processes are not much efficient [1]. Current procedures involves multiple sensors to detect various parameters of the wafer and by analyzing the sensor data on a connected system the system further tells about the authenticity of the wafer. Such kind of numeric data processing requires high level of computing resources. At one end such system requires time and demanding resources to be installed at the facility on the other end they also need to be supplied with high amount of processing power in terms of CPU or GPU further huge amount of RAM is also required to process such big volume of data. Even after that the current systems are not that efficient and time consuming. The novel idea in this paper is to use the recent developments in the field of Artificial Intelligence and Deep Learning to enhance the productivity of wafer inspection facility. This paper proposes a state of the art Deep Learning model which can detect a wafer defect by processing its image using Convolution Neural Network (CNN).

[1 -] Dept. of ECE, Suresh Gyan Vihar University, Jaipur (Raj) India
ORCID ID : 0009-0003-6411-1308
[2] Dept. of ECE, Suresh Gyan Vihar University, Jaipur (Raj) India
ORCID ID : 0000-0002-5452-8658

** Corresponding Author Email: Sharma.pankaj91@gmail.com*

1.1 Deep Learning

Artificial Intelligence, Machine Learning and Deep Learning are the most popular terms in the area of automation today. Deep learning is a subset of Machine Learning which further is a part of Artificial Intelligence. AI is a broader term which brings the human intelligence and human behavior to machine and computerized systems. Machine Learning is a method which learns by using already available data values and used in finding patterns and building predictive modeling, automatic analytic model building etc. Deep Learning works on bigger volume of data involving complex computation at a very high level [12]. It uses multi layer neural networks to process and compute the data to build data-driven intelligent models.

2. Literature Review:

Arnel C. Fajardo et al [1], proposed a CNN model which was trained on different images of defective and non defective PCB wafers and performed efficiently in detecting the kind of defect present in each wafer. The defect classes are spur, scratches, pinholes, mouse bites, excessive conductor, missing conductor, copper track.

Zhijiang Xiong et al [2], proposed a PCB defect detection algorithm based on YOLO v8. This proposed work showed 97% accuracy in defect detection. It has also proposed a defect detection user interface for easy determination.

JunYi Lim et al [3], studied the tradeoff between accuracy and time for the available methods and proposed a new deep learning network which works efficiently for tiny defects on PCB. It is made up of multi scale feature pyramid network and stress upon union loss function. It has achieved mean average precision of 99.17%.

Rey-chue hwang et al [4], used AOI i.e. area of interest based calculations in model where the authors have pressed on automatic optical inspection which has proved to be the best for defect detection. The dataset was taken from industry and the overall model deals with the problem of poor imaging because if poor lighting and shadows. A very few parameters are used in making of model and the accuracy given is 95%, recall is 94%, detection time is 0.027 seconds.

Marcos Antonio Andrade et al [5], proposed a system for PCB defect detection using visual computing and deep learning for production optimization. In this work they have merged traditional computing with newer deep learning technique and the proposed model has achieved greater than 90% accuracy.

Bixian Feng et al [6], worked on variable defects where they have used global information while emphasizing on very small area and diverse style. Defect Detection Transformer is used which is designed with the base of residual swin transformer. They have focused on multi head self attention mechanism for feature extraction. Experiments were conducted on PKU-market-PCB and DeepPCB dataset

Zhewei Liu et al [7], proposed a cost sensitive DL model based on siamese network by using optimization algorithm NSGA-II to check for cost sensitive threshould value. This has produced exciting results in terms of accuracy of prediction. The official predicting accuracy claimed is 97.6% and the shoving average is given 33.32% in training.

Ye Chan Kim et al [8], developed an automated system for wafer defect detection and localize them with accuracy. The images are first scanned with line space patterns and then a deep learning neural network is designed and used to localize the defects. Model is optimized for design variables before training, after that data augmentation is used to enhance the data sample.

After a detailed survey of literature pertaining to already available methods and deep learning models for wafer defect detection it is important to point out that many models are performing efficiently with good accuracy of prediction. But there are still some significant issues and research gaps present which must be addressed.

Some instances of research gaps are as follows:

- Image De-noising – it is important to remove all noise form images before processing them in to the model. The image data with noisy images cannot give better results in terms of prediction. De-noising the images helps the model to generalize effectively.
- Accuracy – achieved accuracy is good but can be improved for better detection.
- Model size- most of the studied models are very heavy in size. They go in GBs a significant improvement is needed to reduce the size of model.
- Computation cost- the number of parameters in the studied models is way too high. It requires high computational cost to process the dataset on certain number of parameters. A light weight model with lesser parameters is required to bring down the computation cost in terms of GPU, RAM and processor.

Problem Statement and Objective

We are provided with semiconductor wafer images from industry and by using these images we have to make a deep learning model which can detect defective and non defective wafers with greatest accuracy. The outcome of this will be a self sufficient deep learning model which can predict defected or non defected wafer within microseconds.

3. Solving Approach- Implementation of Base Model

We already have pre-trained network with us which has been trained on thousand of images and posses the weights and bias data within. **Efficient Net** Algorithm is one such Algorithm which has given good results in image recognition tasks. It is easy to implement with open source availability of resources.

Efficient Net

Efficient Net was created by Google AI research team [13]. It uses many layers of convolution in order to compute the weights and biases. It is based on compound scaling and Neural Architectural Search (NAS) these two things works together in fusion to achieve best performance and resource efficiency. Compound scaling is achieved by bringing down the parameters and Flops (Floating point Operations per Second) on the other hand it uses NAS to create the baseline Architecture [13].

Compound Scaling

There are three types of scaling – Depth Scaling, Width Scaling, Resolution Scaling

Depth $D = \alpha$

Width $W = \beta \wedge \Phi$

Resolution $R = \gamma \wedge \Phi$

$F = DWR = \alpha\ \ \beta \wedge \Phi\ \gamma \wedge \Phi$, where $\alpha = 1.20$, $\beta = 1.10$, $\gamma = 1.15$

Value of Φ comes from Grid Search, F is scaling factor which is used to upscale the network

Because of its automatic scaling feature this Algorithm is performing nicely in terms of Accuracy and Loss. Here we are looking for maximum Accuracy and Minimum Loss for our deep learning model. There are total of 15 Algorithms pertaining to Efficient Net wherein there are 8 belongs to base version and 4 belongs to 2nd version rest to later versions. Each Algorithm has its own unique architecture and different number of parameters, size, Top 1% Accuracy, Top 5% Accuracy as mentioned in the official documents.

Implementation

We have the dataset of images of both kinds defected and Non defected so these images are used to pass through Algorithms. All images are divided into three parts for training set, validation set and test set, each set is further divided into 2 parts as defected and Non defected. These parts are treated as two classes and problem is treated as **"Binary classification Problem"** where Defected = class '0', Non defected = class '1'

Table 1. Accuracy and Loss Results

Fungal Symptom	*Validation Loss*	*Validation Accuracy (%)*	*Test Loss*	*Test Accuracy (%)*
EfficientNetB0	1.05	73.68	1.03	73.68
EfficientNetB1	0.56	73.68	0.48	81.57
EfficientNetB2	1.08	73.68	0.88	73.68
EfficientNetB3	2.18	26.32	2.42	26.31
EfficientNetB4	1.06	73.68	0.46	76.31
EfficientNetB5	1.63	73.68	1.01	73.68
EfficientNetB6	0.36	81.58	0.23	92.10
EfficientNetB7	1.87	63.16	0.67	78.94
EfficientNetV2B0	1.39	73.68	1.15	73.68
EfficientNetV2B1	1.40	73.68	1.24	73.68
EfficientNetV2B2	0.53	76.32	0.13	94.73
EfficientNetV2B3	**0.48**	**76.32**	**0.13**	**94.73**
EfficientNetV2S	2.15	89.47	0.53	94.73
EfficientNetV2M	0.96	73.68	0.88	73.68
EfficientNetV2L	1.62	89.47	4.68	73.68

As shown in Results 15 different Deep Learning models were built using 15 Algorithms of EfficientNet series but none of the above models gave desired results in terms of prediction Accuracy and loss. We are looking for maximum Accuracy and minimum loss, among all the above models only **EfficientNetV2B3** gave best result as Test Accuracy of **94.73** and Test Loss of **0.13**. That is the best result we are able to achieve by this Algorithm.

Limitations of EfficientNet:

1. Resolution – Each of the above EfficientNet Algorithms take a unique size of image as input image within the network ranging from 224*224 up to 600*600. Small image size doesn't provide us maximum features of image.
2. Depth – depth of network refers to the number of convolution layers in the network which are used to find patterns and calculating the convolution, weights and biases within the network. More number of such layers slows down the network it takes longer for the network to train from images as well as more space in memory. Thus it adds to more time and space complexity.
3. Width – number of channels per convolution layer. Wider network slows down the model which adds in shooting up the training time of model.
4. Over-fitting- as seen by observing the performance in most of the models the difference between the Accuracy and Validation Accuracy is large, similarly difference between Loss and Validation Loss is also very high. This becomes more visible in the attached graphical representation of both Accuracy and Loss curves.
5. Time constraint- As the number of layers is very high, the parameters to be trained are in millions, number of channels and filters size is also large. Almost all of these models take a lot of time in training and validation, although these are trained on only 20 epochs

4. Proposed Work:

Proposed work is divided in to 2 parts. Data preprocessing and Deep Learning model building.

As analyzed from the previous results that the Accuracy is not up to the mark and the given dataset contains noisy images which affect the prediction efficiency. The following techniques are used to de-noise the dataset and make it clean for further procedure.

4.1 De-noising the Dataset

First we applied the **Gaussian filter** and **impulse filter** to remove Gaussian noise and impulse noise but the dataset was having blind noise which was too diversified and sophisticated. Then we applied deep learning based filters for better results.

Performance metrics

(1) Peak signal to noise ratio (PSNR) [14] – ratio of maximum signal power to corrupt noise power, usually expressed in log decibel scale.

(2) Structural Similarity Index (SSIM) [14] measure of similarity between noisy image and ground truth clean image based on luminance, Contrast, Structure.

EDA on the Dataset

- Visualizing clean and noisy images
- Mean pixel distribution of images and its analysis using histogram
- Analyzing the PSNR and SSIM value of images
- Creating image patches for further processing

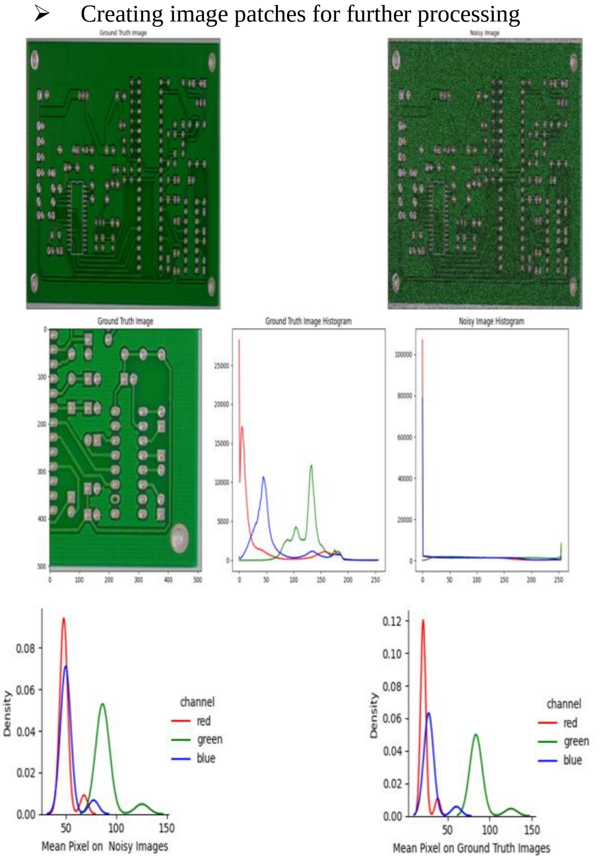


Fig 1. Ground truth image and noisy image, histogram plot for Ground truth image and noisy image, mean pixel plot for both images

De-noising with common filters

Table 2. Deep Learning based models [15][16][17]

Model	*Epochs*	*Loss Function*	*Train loss*	*Test loss*	*PSNR*	*SSIM*
AUTOENCODER	30	MSE	0.0012	0.0012	30.11	0.6
CBDNET	30	MSE+λX	0.00055	0.00054	34.20	0.74
PRIDNET	30	MSE	0.000559	0.000567	34.11	0.72
RIDNET	**30**	**MSE**	**0.000321**	**0.000332**	**35.82**	**0.84**

Table 3. Non local mean algorithm

PSNR pre De-noising	*PSNR post De-noising*	*PSNR Improvement*
17.06485741232191	17.703885994567752	0.6390285822458424
16.970248742272958	17.644692036787575	0.6744432945146173

It is analyzed from the table RIDNET[19] has performed the De-noising task with most Accuracy and best parameters. So we have selected RIDNET model to filter all the image data for further processing.

PSNR range for Ground Truth Noisy image pairs 25DB- 28 DB

SSIM range for Ground Truth Noisy image pairs 0.1 – 0.6

RIDNET model Quantization

Quantization helps in reducing the model size at the same time improve hardware accelerator latency and CPU by reducing the precision of model parameter numbers. This provides us with light model with reduced size and faster computation. We have used pre-trained float tensorflow model to convert the RIDNET model. The results are as follows:

Table 4. PSNR, SSIM and model size for RIDNET and Quantized RIDNET comparison

Model	*PSNR -Test data*	*SSIM- Test data*	*Model size*
RIDNET	35.82	0.84	18.66 MB
RIDNET-	35.98	0.82	7.82 MB

Table 5. Performance of RIDNET-Quantized model on image data samples-

Sample	*PSNR before De-noising (DB)*	*PSNR after De-noising (DB)*	*SSIM before De-noising*	*SSIM after De-noising*	*Prediction time (Seconds)*
Sample1	18.76	32.56	0.34	0.56	1.3
Sample2	22.43	31.44	0.54	0.76	1.5
Sample3	20.21	28.33	0.52	0.81	1.5
Sample4	20.80	33.67	0.32	0.89	1.5

4.2 Proposed Deep Learning Model:

Keeping in mind the limitations of EfficientNet we have designed a unique network of different layers which can provide best Accuracy and minimum Loss. The aforesaid network is named as **TeetuNet,** designed in such a way that it should overcome the limitations of EfficientNet at the same time should perform better than that. So many permutation and combination were tried and tested in deciding the number of layers of convolution layer, number of channels, size of filters etc. At last an efficient network is designed with least number of layers, filter size and channels. In addition to above other different layers are also included in the network.

Salient Features of TeetuNet:

1. **Batch Normalization-** batch normalization improves the efficiency and reliability of network by stabilizing the training process and introducing of generalization in model

Features of Batch Normalization

- Stabilize the training process- by using internal covariate shift
- Higher learning rate- speed up the training of model
- Generalization – normalize the activation of a layer thus reduce over fitting
- Easy initialization – it helps to overcome the sensitivity of model in initializing the initial weights and further training the model.

2. **Dropout layer-**

Dropout layer is installed right after particular layer of Convolution and Dense layer. This layer help us to overcome the over fitting problem. As per the given rate of percentage this layer will automatically drop some neurons from calculation thus making the calculation less complex. Model will also be a little less bulky and fast as the number of neurons drop.

3. **Image Size-** the inputs to network are high quality images ranging from 1500 to 4024 pixel size. The number of features extracted is high thus the efficiency of the model is highest.
4. **Compact–** this network has very less number of convolution layers, channels which makes it less bulky.
5. **Fast-** less number of layers, parameter and introduction of Dropout layer in the network makes it very fast to train and predict. It takes very less time in processing.
6. **Data Augmentation –** Data Augmentation is used to enhance the volume of image data to have more samples in training part [**11**].

Table 6. Model Summary

Layer	*Description*
Convolution 2D	*Filters = 32, kernel size = 5*
Batch Normalization	
Convolution 2D	*Filters = 64, kernel size = 5*
Dropout	*0.25*
Convolution 2D	*Filters = 96, kernel size = 3*
Batch Normalization	
Flatten	
Dense	*Units = 80*
Dropout	*0.25*
Dense	*Units = 96*
Dense	*Units = 1*

Hyper-parameters tuning-chosen from GridSearchCV, a small description of parameters:

(1) *Optimizer*- Adam optimizer is used for weight optimization and minimizes the cross-entropy loss function. Adam(Adaptive Moment Estimation) taken from previous algorithms: RMS prop (Root Mean Square Propagation) and momentum. For every parameter it involves 2 moving averages: mean and un-centered variance

(2) *Epochs* – epochs are set on a maximum of 20

(3) *Batch size* – number of samples in each iteration. Dataset is divided into small batches of fix sizes which id fed into the network for training the model. We have kept it 3.

(4) *Learning rate*- we have kept the learning rate as 0.001. Learning rate is a number which controls the step size during optimization.

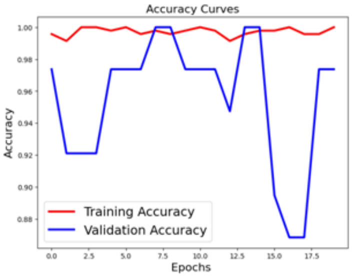
(5) *Sigmoid loss function* – for binary classification sigmoid function is used.

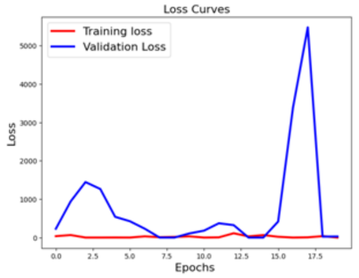
(6) *Dropout value* – set as 20%, used to randomly drop out the neurons from network for better performance.

The available image dataset was fed into this network for training, validation and testing process. This model was trained for 20 epochs. the network got trained very fast and gave phenomenal results in terms of Loss and Accuracy. The detailed results are as follows:

Table 7. Loss and Accuracy Data for TeetuNet

Validation Loss	*Validation Accuracy (%)*	*Test Loss*	*Test Accuracy (%)*
27.63	97.37	0	100





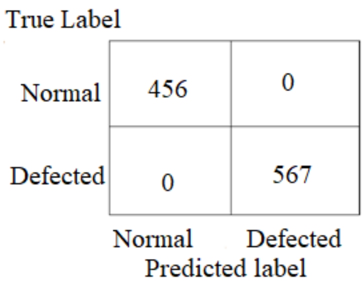


Table 8. Parameter comparison with previous model

Model	*Accuracy (%)*	*Sensitivity*	*Precision*	*F1-Score*	*AUC*
EfficientNetV2B3	94.73	93.78	98.77	96.88	0.98
TeetuNet	100	100	100	100	1

model now can be deployed on the cloud platform and then it can be used to detect defects. More images can be fed into the model and high quality camera equipment can be installed on the production line which can take high resolution pictures of the wafers thus by analyzing those pictures the model can predict the class.

This model has overcome the drawbacks of EfficientNet on parameters such as Depth, Width, Complexity, Over-fitting, Accuracy, Loss, Size, Response time and speed of computation. By comparing to the past versions of other Network and by analyzing the results shown in table 9, we have reached to the conclusion that this model has proved to be the best model till date.

Furthermore efforts can be put in to development of a deep learning model which can go on a step forward and classify the kind of defect a particular wafer has. In second step images can be further grouped on the basis of defect type.

Author contributions

Pankaj Kumar Sharma: Conceptualization, Methodology, Software, Field study, Data curation, Writing-Original draft preparation.

Dr. Sandhya Sharma: Visualization, Software, Investigation, Writing-Reviewing and Editing, Validation.

Conflicts of interest

The authors declare no conflicts of interest.

References:

[1] Arnel C. Fajardo et al, Defect Detection and Classification in Printed Circuit Boards using Convolutional Neural Networks, ICECAA, 2023

[2] Zhijiang Xiong et al, A Design of Bare Printed Circuit Board defect detection based on yolov8, Darcy and Roy, 2023

[3] Jun Yi Lim, Xin Wang, A deep context learning based PCB defect detection model with anomalous trend alarming system,Elsevier BV, 2023

[4] Ray-Chue Hwang, Chun Chen et al, PCB Defect Detection Based on Deep Learning Algorithm, processes, march 2023

[5] Marcos Antonio Andrade, Yuzo Lano, Leandro Rochini Ximennes et al, System for PCB Defect Detection Using Visual Computing and Deep Learning for Production Optimization, IET Circuits, Devices & Systems, 2023

[6] Bixian Feng et al, PCB Defect Detection via Local Detail and Global Dependency Information, Sensors, 2023

[7] Zhewei Liu, Cost-Sensitive Siamese Network for PCB Defect Classification, Computational Intelligence and Neuroscience, 2021

[8] Ye Chan Kim et al, Defect recognition in line-space patterns aided by deep learning with data augmentation, SPIE, 2021

[9] Qianru Zhang, Meng Zhang, Deep learning based solder joint defect detection on industrial printed circuit board X-ray images, Complex & Intelligent Systems, 2022

[10] Logan G. Wright, Tatsuhiro Onodera et al, Deep physical neural networks enabled by a backpropagation algorithm for arbitrary physical systems, cornell university Nature 601, 549-555 (2022)

[11] Jihun Ahn , a Ye Chan Kim et al, Defect recognition in line-space patterns aided by deep learning with data augmentation, SPIE ,Sep 2021

[12] Jungsuk Kim, Jungbeom Ko, Hojong Choi , and Hyunchul Kim , Printed Circuit Board Defect Detection Using Deep Learning via A Skip-Connected Convolutional Autoencoder, Sensors, July 2021

[13] Mingxing Tan, Quoc V.Le, EfficientNet:Rethinking Model Scaling for Convolutional neural networks, ICML, June 2019

[14] PSNR , SSIM , https://ieeexplore.ieee.org/document/5596999

[15] Shi Guo et al, Toward Convolutional Blind Denoising of Real Photographs, CVPR, 2019

[16] Yiyun Zhao et al, Pyramid Real Image Denoising Network, IEEE, 2019

[17]Saeed Anwar et al, Real Image Denoising with Feature Attention, ICCVW, 2019