Machine learning versus Deep Learning in Low Yield Wafer Map Classification

1st Congshu Zhou
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
zhoucongshu@gmail.com

2nd Hingpoh Kuan
Product Engineering
Globalfoundries Singapore
Singapore, Singapore
hingpoh.kuan@globalfoundries.com

3rd Guozhong You
Product Engineering
Globalfoundries Singapore
Singapore, Singapore
guozhong.you@globalfoundries.com

4th Douglas Chan
Product Engineering
GlobalFoundries Singapore
Singapore, Singapore
douglas.chan@globalfoundries.com

5th Jason Khaw
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
jason.khaw@globalfoundries.com

6th Summer Boo
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
summer.boo@globalfoundries.com

7th Hein Mun Lam
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
heinmun.lam@globalfoundries.com

8th Elton Ng
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
elton.ng@globalfoundries.com

9th Juliana Puey
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
juliana.puey@globalfoundries.com

10th Saiking Chong
Yield Enhancement
GlobalFoundries Singapore
Singapore, Singapore
saiking.chong@globalfoundries.com

2021 32nd Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC) | 978-1-7281-8645-0/20/$31.00 ©2021 IEEE | DOI: 10.1109/ASMC51741.2021.9435672

***Abstract* – Wafer maps provide important information in investigating the root causes of low yields during semiconductor manufacturing processes. This paper discusses and compares two methods of wafer map classification, machine learning and deep learning. Wafer map classification is important, because different wafer map defect signatures can be attributed to different root causes for low yield investigation. The automated classification of wafer maps will allow for early defect detection which leads to improved yield enhancement.**

Keywords – wafer map classification, Convolutional Neural Network, Support Vector Machine, yield analysis

I. INTRODUCTION

In the semiconductor manufacturing processes, bin wafer maps are used to visualize yield map patterns and identify process issues. Inline wafer test tools perform yield test after the wafer goes through the entire manufacturing process to monitor for die abnormalities. A wafer map is then created based on the yield result, with test dies being labeled as good/bad bin using different colors.

One of the main purposes of visualizing wafer map is to monitor any abnormal low yield signatures and respond to process problems quickly. A library is normally created storing the yield signatures with their corresponding root causes. With this library, low yield pattern similarities between wafers can be established and be analyzed further for root cause analysis when investigating and solving yield issues. This library can also be used for future query of historical wafer maps with defect root causes.

In this paper we will discuss the use of a Support Vector Machine (SVM) and a Convolutional Neural Network (CNN) for wafer map classification. Our paper is organized as follows. In Section II, we share on the input dataset that is used. Section III discusses on SVM, including feature engineering and model fitting. Section IV follows, describing the CNN model selected for training. The conclusion is presented in Section V.

II. INPUT DATASET

In this paper, we define 12 types of different signatures (a.k.a. defect patterns or classes) for classification (Fig 1). Each of the signatures has around 300-500 images. These datasets are all manually labelled by the engineers during manual sort classification.

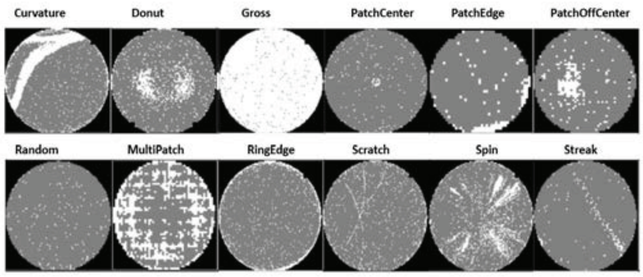

Fig 1: Input dataset

III. MACHINE LEARNING MODEL

In few years ago, due to the lack of computing power and hardware, computer vision projects were usually developed through machine learning. The model was usually achieved through the following steps as Data pre-processing, feature engineering and model fitting. In this paper, in order to keep the picture information as complete as possible, we used the raw image data without any pre-processing. Our most effort lied in the extraction of image features. Before the model training, we extracted the image features through three different methods as followings: (1) local pattern density, (2) Radon transfer, and (3) dominant low yield patterns or shapes. The extracted features are then fed to the SVM model for training.

A. Feature engineering (local pattern density)

We divide the wafer map into 13 areas (Fig 2). In each area, the pattern density is calculated and extracted as a feature, where pattern density is defined as the number of bad die divided by the total die number in that area.

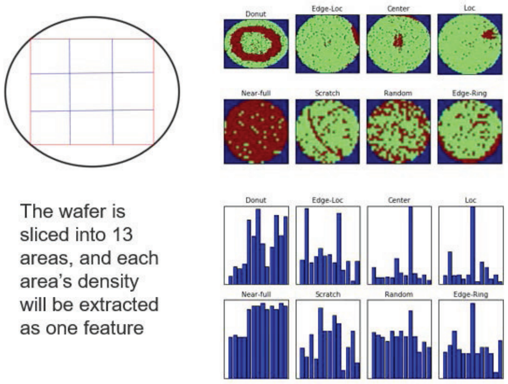

Fig 2: Local defect density

B. Feature engineering (Radon transfer)

Radon transform is applied on raw wafer map data to generate the new features which are exhibiting the geometric information of failure patterns in wafer map. After the Radon transfer, we extracted the mean and standard deviation from the x-axis of the radon images (Fig 3). As is shown in the following pictures (Fig 3), the radon images are quite sensitive to the wafer map signatures such as donut, patch center, patch edge, Gross, etc. The features we extracted could well differentiate those wafer map types.

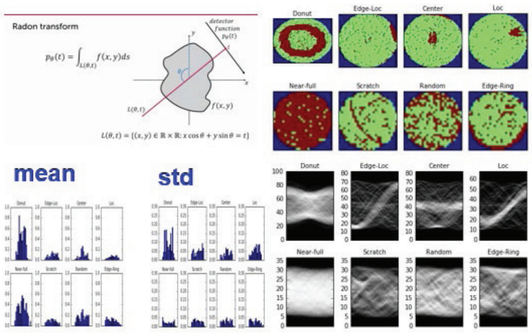

Fig 3: Radon transfer on low yield wafer map

C. Feature engineering (dominant low yield pattern)

Before we extracted the features of dominant low yield pattern, pre-processing on the image is performed. After the pre-processing of the image, only the most majority of the low yield pattern was left, and other low yield patterns were filtered as noises. The features of dominant low yield pattern is sensitive to patch defects, but for streak and scratch the features are not obvious.

In this paper we extracted totally 5 geometry features for this part as show in Fig 4.

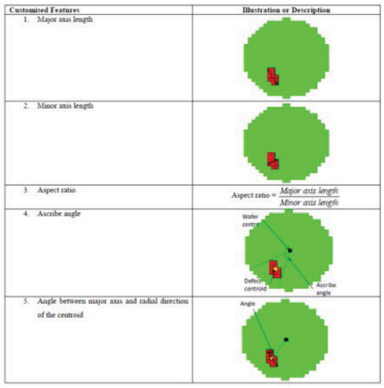

Fig 4: Dominant low yield pattern

D. SVM model fitting

In this paper, the machine learning model we used was SVM. Of course, the other machine learning model such as k-Nearest Neighbors, Decision Trees, Naive Bayes, Random Forest and Gradient Boosting could also be used, and the result we have from those models was quite similar.

The input features of SVM are from the result of feature engineering, 13 features from local pattern density, 40 features from radon transfer and 5 features from major low yield pattern geometry.

Fitting the features into the SVM model for training, we obtain an accuracy of about 59% for 12 different classifications, which is rather unsatisfactory. Fig. 5 shows the normalized confusion matrix. As can be seen, some classification labels yield a very low accuracy, notably edge-based pattern such as scratch (21%) and streak (19%). It is because, the features we captured through radon transfer, dominant defect and local density are not sensitive to edge-based patterns such as scratch and streak.

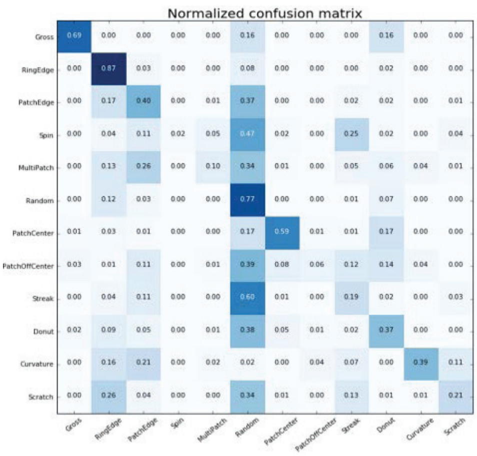

Fig 5: Normalized confusion matrix for SVM model training

IV. DEEP LEARNING MODEL

With the development of times and the progress of technology, we obtained high-performance computing capabilities and hardware including GPU specially designed for deep learning. In this paper we did the CNN (convolutional neural network) deep learning model training by using the GPU of Nvidia Tesla P40.

The convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. Deep convolutional neural networks (CNN) have recently advanced the state-of-the-art image classification performance and became the standard approach for any image classification tasks. Since deep CNN can learn rich features at each layer, these intermediate features are used as good descriptors for image retrieval.

There are many types of CNN structures, such as LeNet, AlexNet, ResNet, VGG19Net, etc. In this paper, we use a simple CNN structure LeNet with 4-level convolutional network.

A. CNN net structure

Fig 6 shows the selected CNN configuration, LeNet. The input wafer map image size is 96 × 96 × 3. We have four convolutional layers with the receptive field size of 3 × 3 and stride 1. The first convolutional layer has 16 channels, while the second, third, fourth convolutional layers have 32, 64, 128 channels, respectively. The rectified linear activation (ReLU) function is used for each convolutional layer, with batch normalization added after the activation. The max pooling size is 2 × 2. The fully connected (FC) layer with the size of 512 is added after the convolutional layers with softmax activation for the class probability calculation.

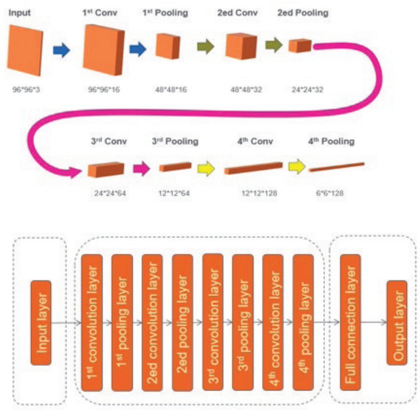

Fig 6: CNN configuration

B. CNN model training

The speed of model training by using GPU is much faster than that of CPU. In this paper, we trained the model totally 120 epochs, at the beginning, the validation loss was not quite stable, the accuracy and loss became smooth after around 70 epochs. Our final training accuracy was around 89.68%.

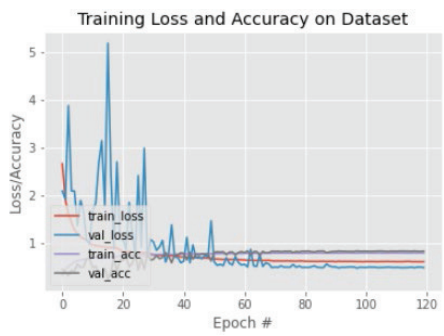

Fig 7: CNN training/validation loss and accuracy

The CNN model confusion matrix is shown in Fig 8. It can be seen that for most defect classes, CNN has a rather satisfactory performance, with an average accuracy of close to 90% for 12 different signatures. There is "confusion" between Patch Edge and Ring Edge though, whose defect signatures look quite similar. Chances are high that these wafer maps are wrongly classified during manual classification (labelling).

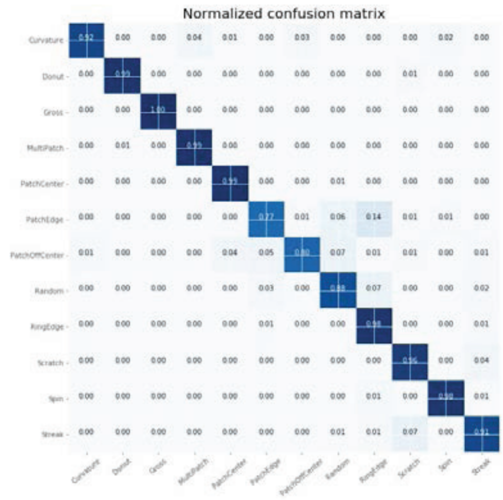

Fig 8: Confusion matrix (CNN)

V. CONCLUSION

In this paper, we compare the 2 methods of classification using SVM and CNN. The classification accuracies of each of the methods and signatures are as tabulated below (Fig 8). Our CNN model yields an average classification accuracy of ~90% which is much higher than that of the result from SVM. The deep learning depends on the capabilities of computing and hardware, if the capabilities allow, CNN is preferred in WMC.

Although the machine learning result was not as satisfied as deep learning, however, in case of the limitations of computer power, machine learning is also an option to choose. If we do not have too many classifications, and the patterns we need to recognize is radon sensitive(such as patch edge, center, off center, donut, gross and ring edge), the feature engineering and followed by machine learning is also a good choice.

	Machine learning acc	Deep learning acc
Curvature	39%	92%
Donut	37%	99%
Gross	69%	100%
MultiPatch	10%	99%
PatchCenter	59%	99%
PatchEdge	40%	77%
PatchOffCenter	6%	80%
Random	77%	88%
RingEdge	87%	98%
Scratch	21%	96%
Spin	2%	98%
Streak	19%	91%

Fig. 8. Accuracy comparison between machine learning and deep learning

ACKNOWLEDGMENT

I wish to thank the following for their contribution to this paper; Hing Poh, Guozhong, and Douglas for their valuable participation in providing the wafer map dataset for model training; Hein Mun Lam, Jason Khaw and Summer Boo for their help in defining and building the business and automation flow; Elton, Juliana, and Sai King for their support in the infrastructures which are used for model creation, training and deployment.

REFERENCES

[1] Takeshi Nakazawa, "Wafer Map Defect Pattern Classification and Image Retrieval Using Convolutional Neural Network", IEEE Transactions on Semiconductor Manufacturing (Volume: 31, Issue: 2, May 2018 DOI: 10.1109/TSM.2018.2795466)

[2] Cheng Hao Jin, “Wafer map defect pattern classification based on convolutional neural network features and error-correcting output codes”, Journal of Intelligent Manufacturing volume 31, pages1861–1875(2020)

[3] Jianbo Yu, "Wafer Map Defect Detection and Recognition Using Joint Local and Nonlocal Linear Discriminant Analysis", IEEE Transactions on Semiconductor Manufacturing (Volume: 29, Issue: 1, Feb. 2016 DOI: 10.1109/TSM.2015.2497264)