Detecting Transistor Defects in Medical Systems Using a Machine Learning Approach

Husnu Murat Kocak
Department of Computer Engineering
Galatasaray University
Istanbul, Turkey
murathkocak@gmail.com

Ahmet Teoman Naskali
Department of Computer Engineering
Galatasaray University
Istanbul, Turkey
tnaskali@gsu.edu.tr

Jerome Mitard
Compute and Memory Department
IMEC
Leuven, Belgium
mitard@imec.be

***Abstract*—The correct and flawless functioning of medical devices is of utmost importance for patients and the medical sector. Although medical device manufacturers utilize various protocols and procedures during the development and production of medical devices, the integrated circuits are usually manufactured by other parties. During the manufacture of the semiconductors utilized in the integrated circuits, the performance of the components can vary from batch to batch and even samples within the same batch can have different performance characteristics. For applications in life critical applications the selection of the most fit for purpose components is vital. In this paper, we propose a novel method for the semiconductor industry to verify the quality of transistors. The I-V graphs of the components are evaluated by multiple Convolutional Neural Networks (CNN) using visual data similar to expert evaluation, then these Machine Learning (ML) architectures utilize a multi model ensemble technique where one architecture providing a negative output will overrule the vote of the other architectures is utilized to assure very stringent quality control. Our method is tested on CMOS transistors and the results are comparable to those of experts with 10 years of experience in the industry.**

***Index Terms*—Machine learning, Convolutional Neural Networks, Big data, Medical systems, Transistor Defects, Semiconductors, Multi Model Ensemble**

I. Introduction

Most high technology medical devices utilize integrated circuits of one form or another. The quality and performance of these components are of utmost importance for medical devices. These integrated circuits that allow us to make precise calculations consist of complex and interrelated transistor components. After a long period of design and modeling process, with the development of production tools and fab equipment, extensive researches are being done to reduce the size of the transistors in order to gradually increase the number of transistors contained in the chips. Nowadays, 7nm silicon logic architectures are widely used, while the most advanced manufacturing facilities can reduce it down to 3nm [20].

The demand for semiconductors keeps growing and seeing an uptake as more industries are transforming digitally, leading chip makers and governments to build more and more capacity into supply chains for electronics products which are more and more innovative. Without special attention, this push may have consequences on the quality control of the final electronics products. On the R&D front, an increased use of new materials and new device architectures are seriously considered nowadays [19] to sustain the rapid performance growth of the compute-oriented system. New physics-based mechanisms or process-induced phenomena should be captured as early as possible from a massive amount of experimental data. Not doing so may close the door to disruptive, smart, sustainable solutions in multiple domains such as healthcare, clean energy, or Industry 4.0 to name few.

One tool which can used to quickly, accurately and predicatively impact awareness of the semiconductor devices is based on Machine Learning (ML) and Neural Networks (NNs) through relationship between inputs and outputs and predicts outputs with respect to new inputs. The use case which will be given in this paper is the automatic classification of output characteristics of the key building blocks of the semiconductor industry: the Metal Oxide Semiconductor (MOS) transistor. Hence, after a rapid description of the data to be analyzed, we will place this work in the context of the similar research papers and we introduce our new approaches bases on.

The Metal Oxide Semiconductor Field Effect Transistor (MOSFET) is a semiconductor device that is widely used for switching purposes and for the amplification of electronic signals in electronic devices. It is a four-terminal device having source (S), gate (G), drain (D) and body (B) terminals and its functionality depends on the electrical variations happening in the channel width along with the flow of carriers (either holes or electrons). If the charge carriers which enters into the channel through the source terminal (Is) are not properly exiting via the drain (Id), the MOSFET is considered as defective or non-healthy. Fig. I represents some examples of I-V (electric response) curves taken from experimental routines of transistors. Respectively, green plot represents current of gate, red for current of drain, blue for current of source, yellow for current of body and dashed plots represents first routine while straight line plots represents second routine. As a single experiment may not characterize the component properly, a secondary experiment with a different voltage level and the two results are interpreted together.

In this work, the inputs characteristics are coming from a Research and Development line. However, the materials used in the MOSFET are conventional coupled to an advanced N14 FinFET architecture.

For quality control for these components, currently some

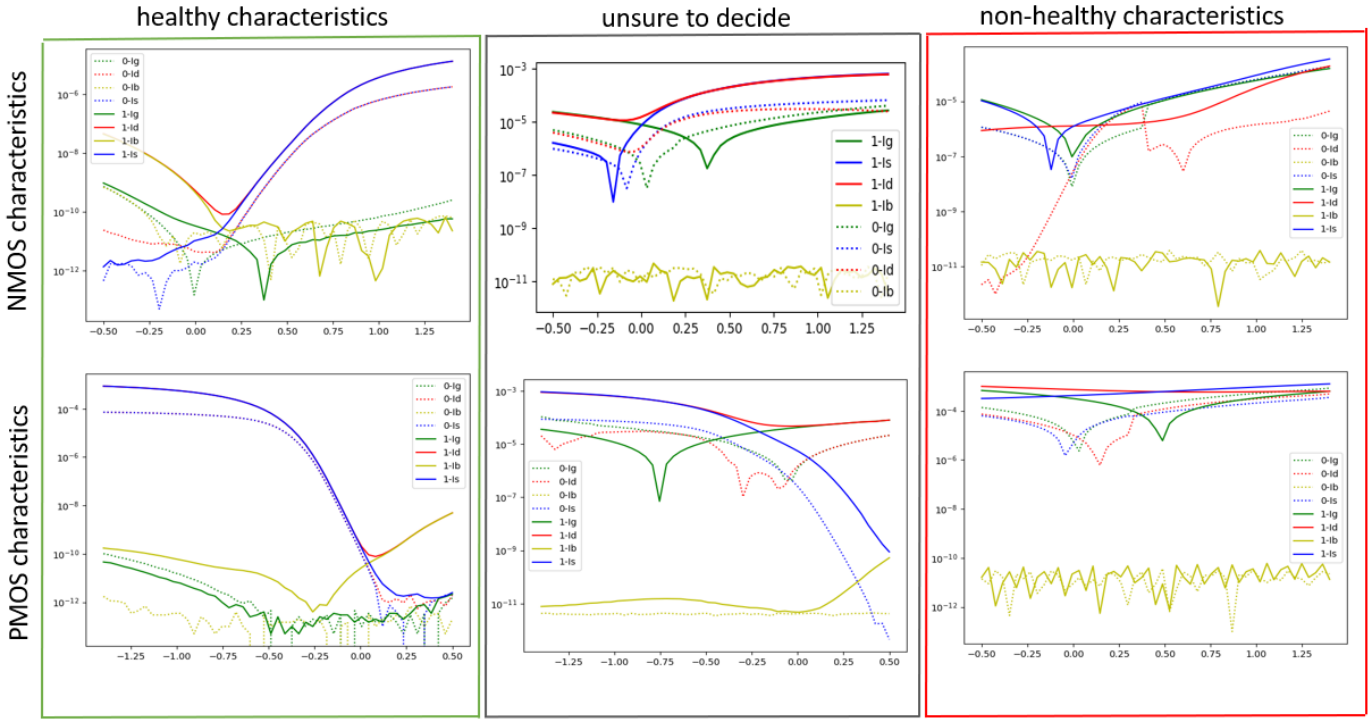

Fig. 1. 6-terminals I-V characteristics of a FinFET and at two inputs drain voltages (V_{ds}) (0 is V_{ds}= 50mV, and 1 is V_{ds}= 800mV). Labels of healthy, non-healthy and unsure made by the same expert.

samples are extracted from each batch which are tested by expert evaluation. Expert evaluation in this field is by no means a gold standard as even the same data evaluated at different times by the same expert can yield different results. These factors necessitate multiple evaluation of the same data which is a tedious and costly process. Automated methods as the one provided in this article can enable the rapid and accurate testing of all of the components manufactured.

The structure of this article is as follows: In the section II, previous attempts at machine learning are described, in the section III is described. In the section III-A the structure of the model is presented. In the section III-B, the data input to the system is described. In the section IV the results are presented, In the section V the possible future works are described, and finally the conclusion of the paper is presented in section VI.

II. Related Work

Early works of Machine Learning (ML) based transistor characteristics categorization had to work with very limited resources. The number of applicable ML algorithms was limited and neural network techniques were not capable of learning very complex tasks due to the lack of computational power and tool support, resulting in the usage of a lower number of neurons and a lower number of layers. Therefore, in order to improve ML accuracy and augment its performance, domain knowledge was utilized in several works. While the incorporation of domain knowledge to the creation of NNs improves their accuracy, this form of manual intervention to the function of a NN deprives it of many of its benefits (e.g. generalizability, autonomous operation).

In [10], the authors have proposed one of the early ML approaches to transistor modelling. The key idea is the prediction of current voltage curve using only 12 neurons bounded most likely by computational memory. Since it is working with very limited memory, some domain knowledge (e.g. monotonicity) was coded inside the model during training process to increase accuracy. Another approach of transistor modelling and classification using neural networks is presented in 2017 [11]. Physical dependency information and transistor specific knowledge is used to train a NN model with 15 neurons. In [12] 2 separated NNs are utilized, one for V_{ds}, another for V_{tg} (top gate voltage) each possessing less than 10 neurons. In other words, since their NN models consist of a limited number of neurons, this limited number of neurons is not sufficient to learn complex information and the system requires the manual coding of domain knowledge which limits the capabilities of the ML.

A genetic algorithm combined with NNs to find best architecture of NNs is proposed in [14] in 2016. However, the usage of this ML model is a very limited since it is only applied to very specific transistor characteristics which would penalize the accuracy and performance should the same technique be implemented to another type of transistor.

Compact model of transistor usually described as the interface to bring a transistor implementation into the simulators. There are commercial compact modelling tools that use ML techniques and classification based on compact models. One of

these is Silvaco TechModeler [17]. Due to limited availability of information, we could not compare the results to ours.

[16] describes of usage a random forest algorithm with domain knowledge for defect detection. However, this approach could be a drawback in this area of application. The nature of the random forest algorithm is not very convenient for updating itself. [6] presents an NN with 250 neurons without domain knowledge for predicting signal outputs and categorizes transistors based on outputs. Using such NNs architectures puts challenges on the scalability of the system. Furthermore, the data needs to be preprocessed which adds additional limitations in the training phase because limitation of data reduces the range of variants and requirement of scaling data creates conflicts between real-world and reprocessed data as is pointed out by the authors.

Besides, our CNN approach has wide use for anomaly and defect detection and classification. Some examples that use CNNs as defect detector are: in [18] where authors detect damages in roads and propose remarkable accuracy results compared to state-of-art techniques. In another example, defect detection in X-ray images with state-of-art CNN models is performed, fed by transfer learning in [21] and they present that a CNN model can find defects in under 1 second. Our approach has similarities with the stated works. However, in this work, we have focused on finding ways of detecting defects on transistors in a nanometric scale, and we augment it with a multi model ensemble technique by combining multiple CNN models. Due to the requirements of medical devices, this multi model ensemble is biased towards classifying only the devices that are healthy.

In summary:

- We propose here a ML model with increased sensitivity for internal circuits of medical systems that operate in very vital places. This is very critical due to humans life relies on it.
- We explore how our approach can work on limited data sets such as limited experimental.
- Because domain knowledge is not incorporated in to the model, the solution is more generic and is capable of working with devices ranging from very conventional up to very disruptive ones.

III. Our Machine Learning Method

In this paper, we present a Convolutional Neural Network (CNN) based machine learning model to learn from I-V curves and classify transistor characteristics based on curve information to detect leaks and possible defects. If the CNN model can accurately classify transistors as healthy (no or minimum leaks) or non-healthy (leak current above the acceptable threshold), then it is sufficient to use as pass/fail checker in the quality control process due to faster response.

The lack of expertise in humans manifests itself as increased false positives in the classification of the data and this may cause misdirected results during the decision-making phase that we have mentioned in the previous sections. Along with our ML approach, the system was also designed to provide double-check control to the experts as an auxiliary tool that informs the expert when they decide against the results generated by our model.

This work is done in the scope of a major project which aims to achieve full autonomous quality control during the fabrication process of transistors in chip with machine learning techniques. With regards to this work, our aim is to detect the defects in transistors given their I-V curves.

A. Model Setup

Given that in the current state of affairs, the IV graphs of the transistors are evaluated using humans, human vision and previous experience to be more precise, we chose an approach that mimics experts reasoning to globally classify good or bad IVs transistor characteristics. This approach does not strictly encode any domain knowledge into the model. For the task of detecting transistor defects we chose a CNN architecture which has proven its efficiency in pattern recognition and image classification tasks.

Our CNN-based ML transistor characteristics classifier model is coded in Python and using tensors in Tensorflow 2 framework for ML functionality. As a model architecture we use multi-layered CNNs with a preprocessing layer for scale the input data to predefined size and depth, a data augmentation layer for reducing over-fitting probability and force the model to find features, two connected Conv2D layers with Maxpooling, Dropout layers with Parametric Rectified Linear Unit (PReLU) [7] activator, a flatten layer to transform 2D features into 1D shape, a dropout, and a dense layer with softmax activator to get probability result. We used ADAM optimizer and Sparse Categorical Crossentropy as loss metric.

B. Data Reprocessing

Experimental I-V curves from routines of transistors labelled and well validated means curves that cause confusion are also excluded by the experts in field are used. Entire dataset consists of 3200 curves, while 70% of dataset is separated for the training process (2560 samples), and then 30% of total curves (640 samples) is reserved for validation. For each iteration, we choose k-fold cross validation technique to generate validation data and create a different validation dataset. The entire dataset is arranged in binary classes which mean half of it are labelled as healthy, other half is labelled as non-healthy to prevent an unbalanced data problem.

Training the model where the data has multiple orders of magnitude causes problems. While some of difficulties are already mentioned in [6], the reprocessing step, thus the range of values is more equally weighted.

Memory management is an important consideration when training neural networks for machine learning. Optimizing memory use not only decreases training time but also can be beneficial for avoiding problems like overfitting. One of the most common techniques to reduce memory usage is to reduce the dimensions of the inputs. Bi-linear interpolation [13] was applied to resize the images to 120px, 160px to obtain a 16 fold reduction. To further reduce memory usage, the images

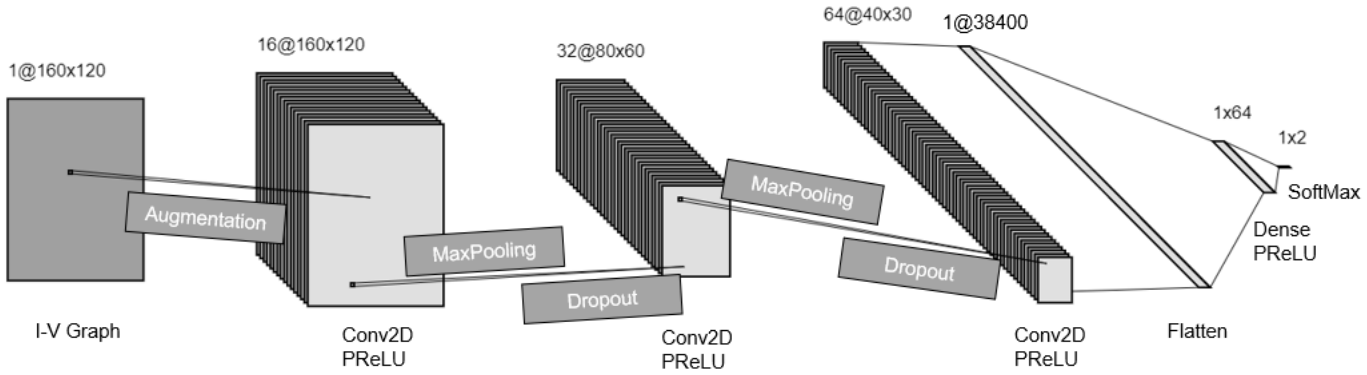

Fig. 2. Visualisation of our CNN architecture that presented by NN-SVG [15] technique

were grayscaled to further reduce the dimensions by another 2/3.

C. Prevent Overfitting

The overfitting problem is when a neural network memorizes the training data instead of finding information features, due to the lack of variety of data. As a result of this, the model performs perfect accuracy on training data, while performance on unseen test data is poor. Overfitting problem first pointed out in [2] as "...the principle of parsimony, calls for using models and procedures that contain all that is necessary for the modeling but nothing more. Overfitting is the use of models or procedures that violate parsimony...". Types of overfitting and their solutions are mentioned in [3]. We used data augmentation, dropout layers and early stopping for the prevention of overfitting.

Overfitting manifests itself as a model with poor generalization. Generalizability refers to the performance metric difference of a model when evaluated on training data (the data it has previously seen) versus testing data (the data it has never seen before). One of the methods to improve generalization is data augmentation. The augmented data will represent a more comprehensive set of possible data points, thus minimizing the distance between the training and validation set, as well as any future testing sets [4]. The CNN model we propose in this work, starts with a data augmentation layer. In this layer, at least one of the following image manipulation techniques are applied to the images that are randomly selected from training data: flip (both horizontal and vertical), rotate, zoom-in or zoom-out, and crop random (5px, 5px) piece of image. As a result, we focused on the model not learning unnecessary features of the image and forcing it to learn the shape feature in IV curves. In addition, we have increased the variance of our data that makes it difficult for the model to memorize the data and allows it to perform more accurate predictions during testing.

Dropout, first introduced in 2014 [5] is a simple but effective overfitting prevention technique in neural networks. The basic idea is based on temporarily disabling some units randomly and their connections in hidden and visible layers in every epoch (number of passes of the entire training dataset the ML algorithm has completed), thus the trained neural network becomes a union of thinned networks with extensive weight sharing, where each thinned network is trained on at least once with all the data. In our network, we combined Conv2D-Maxpooling-Dropout layers sequence three times and one dropout after dense layout that results after every maxpooling and dense layer (extending feature maps while reducing size of image) we create new thinned network with a certain fraction rate of inputs as depicted in Fig III-A.

Lastly, we set callback points to save weights of the complete network and record validation loss after every epoch. After validation loss stops decreasing, to prevent overfitting, we stop training and reset the weights to the latest callback that generates the lowest validation loss. While validation loss indicates how well the features from training data fit to new data, other metrics can be used such as accuracy and training loss.

D. Choosing the right activation functions

Artificial Neural Networks (ANN) are one of the deep learning algorithms and multi-level representation techniques that aim to represent the raw inputs with higher levels of abstraction. Many of these transformations producing learned complex functions. With the development of computational power of processors, more complex and multi-layered neural network models that can produce more accurate results in tasks such as regression, classification and clustering have started to emerge. However, the crucial issue of using deep neural network architectures is the difficulty of developing algorithms to effectively learn the patterns in the data and studies on these issues related to the training of neural networks, are a key research area so far.

Rectified Linear Unit (ReLU) function is the one of the keys to recent successes of neural networks. Several studies [7], [8] show us ReLU over-performs sigmoid, linear and hyperbolic functions and generate better accuracy due to their vanishing gradient problem. Although ReLU has these advantages, it can still be improved due to problems such as dying ReLU problem [9] since our CNN model learns from I-V curve

images. Parametric ReLU (PReLU) multiplies the negative input by a small value (slope parameter learned during back-propagation) and keeps the positive input as is, instead does not zero out the negative inputs as regular ReLU does. As an activation function of Conv2D layers and Dense layers in our architecture, we decided to use PReLU for these benefits.

IV. Experimental Results

In this work, we have tested our CNN model with IV curves from mos_vtk_i4 routine of three different wafers, which each one includes 4176 samples. All training and test data generated by sweeping input signals with a certain step size in SPICE simulations. All input parameters and their corresponding outputs signals are collected in tree structured Json formatted files to serve. Fig. IV shows training and validation loss over epochs. We see that after thirtieth epochs there is a sharp decrement in loss and they are very close to 0 after the seventieth epoch, meaning the CNN model was able to match the features with low loss.

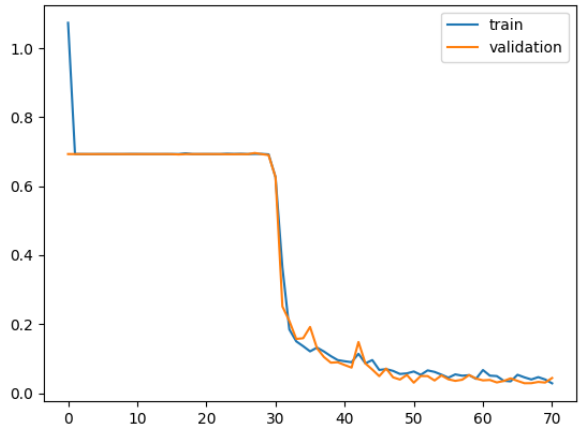

Fig. 3. Training and validation loss over epoch while y-axis represent loss and x-axis represent number of epochs.

Table IV shows the accuracy, precision, sensitivity and f1 scores over three different wafer and mean of all wafers after 70 epochs training which took 12h depending on the number of input samples. While longer training time can overcome the difficulty of finding features with more processing power, it does not guarantee continuous improvement of model. After 70th epoch no improvement was seen. Therefore, Training process was finished at the 70th epoch.

Categorizing the FinFET characteristics relies on expertise in this domain and experts can find different features on curves with years of experience and categorize them accordingly. When these categorizations are compared, although they show huge similarity to a great extent, there may be signal outputs that we can describe as gray areas, which is open to debate regarding to the presence of a a clear defect. Therefore, It was needed to compare our CNN model with experts with 3 different levels of experience with the same test set. In the Fig. 2, the results of the experts and CNN model for the I-V curve categorization taken from 3 different wafers can be seen. Experts 1, 2 and 3 have approximately 5, 10, and 1 years of experience in their fields, respectively. When the results are investigated, it is seen that single CNN model can categorize like an expert of 1-5 years in the field.

TABLE I
Confusion matrix score of CNN model, including mean of all test cases.

Wafer Index	Accuracy	Precision	Sensitivity	F1 Score
18	0.832	0.971	0.554	0.705
17	0.805	0.980	0.536	0.692
14	0.893	0.989	0,625	0.765
Mean:	0.843	0.980	0.572	0.721

A. Improvement of Decisions with Multi Model Ensemble Technique

TABLE II
Confusion matrix score of our multi CNN model ensemble approach.

Wafer Index	Accuracy	Precision	Sensitivity	F1 Score
18	0.903	0.989	0.633	0.771
17	0.871	0.996	0.591	0.742
14	0.940	1.000	0.716	0.834
Mean:	0.904	0.995	0.647	0.782

Instead of relying on a single model for the classification of the components, as the detection of substandard components can is vital for the medical sector, an approach using multiple models was investigated. This approach mimics the quality control of critical systems where more than one expert evaluates the same data.

For these reasons, 3 CNN's were trained and a consortium of CNN's is formed. The input data is evaluated by the models and each model casts its vote. A majority vote for this application was not chosen as a viable option as on one CNN claiming that the component is not fit for purpose is enough to discard the component. As the problem necessitates certainty only unanimous votes were considered as a final result, non unanimous votes were labeled as unsure. This approach increased the reliability of the transistor classification results.

Utilizing the proposed multiple CNN model technique we reached accuracy comparable to an expert with 10 years of experience in the field. In the Fig. 4, we present the comparative accuracy performances of our single CNN model, multi model ensemble technique and, 3 experts in this domain. Although there may be changes in the ratios according to the wafer type, when we examine the average, it can be seen that our multi model ensemble technique generates 90% accuracy, and surpasses the experts that have 1 (85%) and 5 (85%) years of experience, while it is comparable with the expert has 10 years experience (92%).

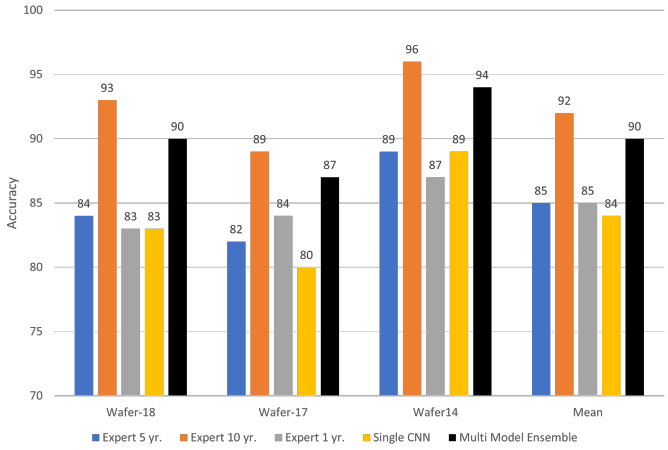

Fig. 4. Comparative accuracy results of experts and our CNN architectures.

V. Future Work

The defect classifier multi model ensemble described in this work is accurate enough and can be evaluated fast enough to be useful in a real manufacturing setting. However, we intent to make more precise selections by combining our current multi model ensemble technique with different ML architectures. At the same time, the research of performance measurements of healthy labelled transistors and their results, can lead way to improvements of experimental devices. Future work could focus on developing a standardized multi model ensemble method to enhance accuracy and scale the performance of defect-free transistor designs.

VI. Conclusion

In this work, we presented an ML approach, more specifically a CNN model that is able to categorize the CMOS transfer curves to defected or not-defected with high accuracy specific to certain transistor characteristics. Using a multi model approach, even when one of the models in the ensemble deems a transistor to be non-healthy, it is classified as non-healty and can be viewed by experts in later stages of the process. Without any changes to the CNN, it can successfully learn to categorize different types of transistors, based on their I-V curves. In the light of the presented results, it is promising that machine learning techniques will be developed to use in terms of both reduction in time cost and increase in detection accuracy of defects that have occurred or will occur.

An ML solution is useful not only for the decrease in development times but also it can be used as a more centralized part of the decision making process to help clear up confusion among experts. At the same time, due to its fast response, it can be very useful in terms of verification if there is any counter-labeling situation between the expert and the CNN. With these benefits at hand, ML is promising an increment in customer acceptance due to easier access and accelerated technology development.

References

[1] Nwankpa, Chigozie, et al. "Activation functions: Comparison of trends in practice and research for deep learning." arXiv preprint arXiv:1811.03378 (2018).

[2] Hawkins, Douglas M. "The problem of overfitting." Journal of chemical information and computer sciences 44.1 (2004): 1-12.

[3] Ying, Xue. "An overview of overfitting and its solutions." Journal of Physics: Conference Series. Vol. 1168. No. 2. IOP Publishing, 2019.

[4] Shorten, Connor, and Taghi M. Khoshgoftaar. "A survey on image data augmentation for deep learning." Journal of Big Data 6.1 (2019): 1-48.

[5] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." The journal of machine learning research 15.1 (2014): 1929-1958.

[6] Klemme, Florian, et al. "Modeling emerging technologies using machine learning: challenges and opportunities." Proceedings of the 39th International Conference on Computer-Aided Design. 2020.

[7] He, Kaiming, et al. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification." Proceedings of the IEEE international conference on computer vision. 2015.

[8] Ramachandran, Prajit, Barret Zoph, and Quoc V. Le. "Searching for activation functions." arXiv preprint arXiv:1710.05941 (2017).

[9] Lu, Lu, et al. "Dying relu and initialization: Theory and numerical examples." arXiv preprint arXiv:1903.06733 (2019).

[10] P. B. L. Meijer, Neural network applications in device and subcircuit modelling for circuit simulation. Philips Electronics, 1996.

[11] L. Zhang and M. Chan, "Artificial neural network design for compact modeling of generic transistors," Journal of Computational Electronics, vol. 16, no. 3, pp. 825–832, 2017.

[12] M. Li, O. ̇Irsoy, C. Cardie, and H. G. Xing, "Physics-inspired neural networks for efficient device compact modeling," IEEE Journal on Exploratory Solid-State Computational Devices and Circuits, vol. 2, pp. 44–49, 2016

[13] William H.; Teukolsky, Saul A.; Vetterling, William T.; Flannery, Brian P. "Numerical recipes in C: the art of scientific computing (2nd ed.)." New York, NY, USA: Cambridge University Press. pp. 123-128. 1992

[14] K. Lamamra and S. Berrah, "Modeling of mosfet transistor by mlp neural networks," in International Conference on Electrical Engineering and Control Applications. Springer, 2016, pp. 407–415.

[15] LeNail, Alexander. "Nn-svg: Publication-ready neural network architecture schematics." Journal of Open Source Software 4.33 (2019): 747.

[16] Teo, Chea-Wei, et al. "TCAD-enabled machine learning defect prediction to accelerate advanced semiconductor device failure analysis." 2019 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD). IEEE, 2019.

[17] T. Blaesi, "Spice model generation by machine learning," 2019. [Online]. Available: https://silvaco-blog.com/circuit_simulation_modeling/spice-model-generation-by-machine-learning-html/

[18] Pham, Vung, Chau Pham, and Tommy Dang. "Road damage detection and classification with detectron2 and faster r-cnn." 2020 IEEE International Conference on Big Data (Big Data). IEEE, 2020.

[19] N. Horiguchi, Z. Tokei, "A View on the Logic Technology Roadmap", 2020, pp. 12, [Online]. Available: https://online.publicationprinters.com/html5/reader/production/default.aspx?pubname=&edid=a58da11a-3cca-46dd-930c-ed28ab9ac1ff

[20] Radamson, H. H.; Zhu, H.; Wu, Z.; He, X.; Lin, H.; Liu, J.; Xiang, J.; Kong, Z.; Xiong, W.; Li, J.; Cui, H.; Gao, J.; Yang, H.; Du, Y.; Xu, B.; Li, B.; Zhao, X.; Yu, J.; Dong, Y.; Wang, G. State of the Art and Future Perspectives in Advanced CMOS Technology. Nanomaterials 2020, 10, 1555.

[21] Ferguson, Max, et al. "Automatic localization of casting defects with convolutional neural networks." 2017 IEEE international conference on big data (big data). IEEE, 2017.