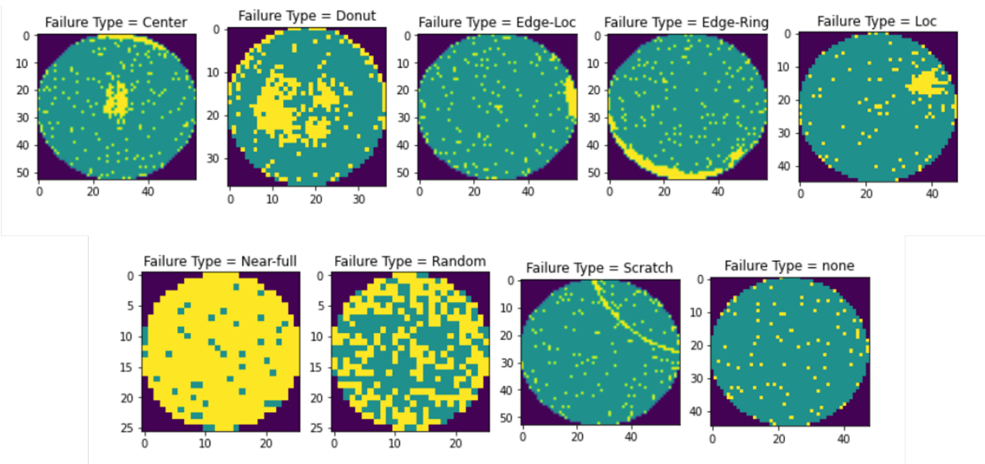
WAFER BIN MAP RECOGNITION
Author : Océane GÖRKE
Course : Computational Intelligence Methods
Dataset : LSWMD.pkl

WAFER BIN MAP RECOGNITION - Augment dataset and train CNN classifier

In manufacturing process, one of the steps of the Integrated Circuit works as testing wafers which are the result of an electrical die-sorting test. This is usually done with semiconductors.
These tests are done in order to provide information on which bins failed and on what tests. These can also determine the performance of the semi-conductor that has just been tested.

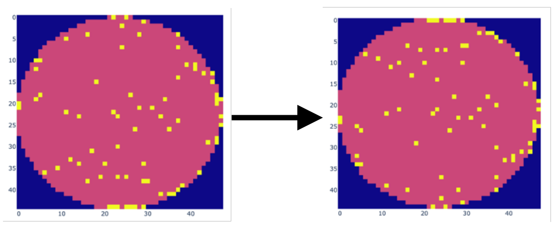
Each bins can have different scores according to the test, usually we focus on 0 and 1 values. After doing a few tests, some failure patterns have been registered. Here is an idea of what they could look like :



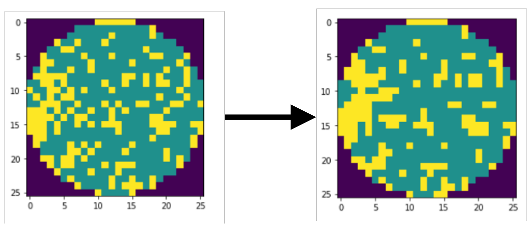
So, the point is to create a neural network model which will be able to recognise a particular pattern by itself. Indeed, each failure type can be seen as a category. This could help the industries to find what the problem could be as each test results from a particular failure.

The neural network used is a Convolutional Neural Network. It actually is a method in computer vision applications that is used to analyse visual imagery. An input image is actually analysed in some parts which are all assigned to a weight value that will be used in the further layers.

But first of all, the data on which the work is based has to be a cleaned data. Also, some data augmentation can be interested to do since it can happen that the dataset is a lot reduced and is not sufficient anymore for the models.
Data augmentation can be done by using a convolutional encoder/decoder on the actual dataset or by just flipping the images already existing.



flipped image

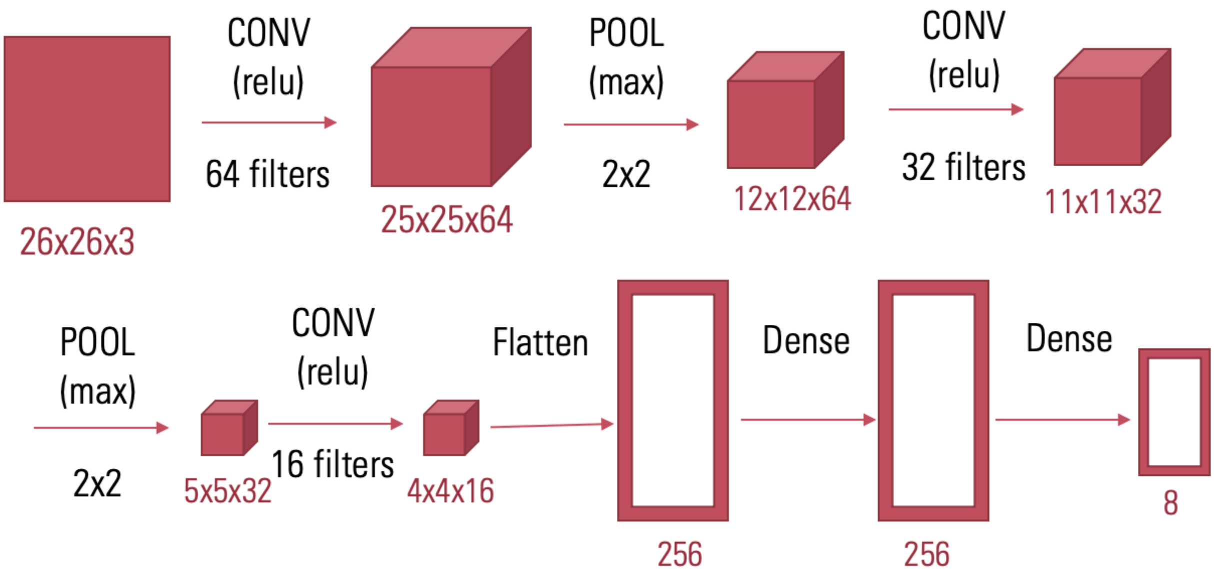


encoder/decoder image generated

A convolutional encoder/decoder is a neural network using convolutional layers that can generate some data similar to the ones in input by adding some noises and modifications.

Then, having this new data, the work can be focused on the implementation of a neural network.

A Convolutional Neural Network can be built in different ways: by computing different combinations of simple or complex layers. In order to find the best one predicting the failure type with the most precision possible, different CNN have been tried. Each time, the loss and accuracy values have been stored and compared.
The loss value results from the mean squared error which is the mean of squared difference between the actual values and the expected ones. What's more, the accuracy is a statistical measure showing the proportion of correct predictions.
So, in order to consider the best model possible, the loss had to be minimised for the greatest accuracy it could be.



Finally, for a resulting loss of 0.1764 and an accuracy of 0.9646, this model below has been chosen as the best one.
Each step represented above has a special role.
The convolutional layer is the first layer dealing with the inputs in order to extract the various features. A ReLU activation function is used. This function computes the maximum between the input values representing the image and 0. It is useful in filtering information that is propagating through the network especially in the convolutional layer where nonlinearity can be.
Usually, a convolutional layer can be followed by a Pooling layer which computes the maximum or the average value from the previous network values divided in n x m dimensions. Here the image is divided in 2 x 2. It can also help to reduce computational costs and preserve the detected features.
The filter parameters are decreased at each step in order to have more precision on the output.
The Flatten layer is a really important layer in classification because it helps object detection and image recognition therefore it was needed here.
The Dense layers are at the end and are useful since it allows to put the activation function to the output. These layers are also necessary in a multi-class classification which is the case here.
The final length of the network is 8 because it the number of patterns destined to be predicted (the none label is not considered here neither as a pattern).

So, this is a global description and explanation of what a CNN could look like and how it can be useful to recognise patterns in a set of images.

In order to go further in the work and the analysis, some comparisons could be done on the network such as a confusion matrix comparing the patterns and their occurrence. Also, testing the model on a complete different dataset of wafer maps could be interesting to see the results.