POLITECNICO
MILANO 1863

Quality Data Analysis

(Prof. Bianca Maria Colosimo)

TEAM n°43

Design of a statistical method for detecting defects in 3D printed fan covers by using images collected in-line.

Authors:

Angeloni Lorenzo lorenzo.angeloni@mail.polimi.it

De Rocchi Luca luca.derocchi@mail.polimi.it

Lalli Ettore lalli.ettore@mail.polimi.it

Longobardo Mattia mattia.longobardo@mail.polimi.it

Academic Year 2022-2023

1. Introduction

The primary objective of this project is to implement the best method for identifying imperfections in PC fan covers. These are produced by using 3D printing technology and are composed of a plastic polymer material known as PA12. In 3D printing, the final components are built layer by layer, resulting in the possibility of encountering several issues at different stages. These problems may lead to the manufacture of defective components. Our aim was to develop a methodology that is able to monitor the quality of conformance in the process. To do so, several stages were needed.

Firstly, we needed to gather data, and to fulfill this objective we went on-site at MADE; in the competence center there is a line where parts move on trays, pass under a camera, they stop at the desired place, and are photographed. The pictures made are then sent to a computer, where they are processed in order to be used for populating our dataset.

Afterword, it was needed to identify which are the anomalies that characterise a non-conformant component. This project focuses on 4 main types of defects, shown in Figure 1: Missing material along the border, Missing material along the corner, Broken/missing struts (the small parts in the net), and Severe lack of struts.

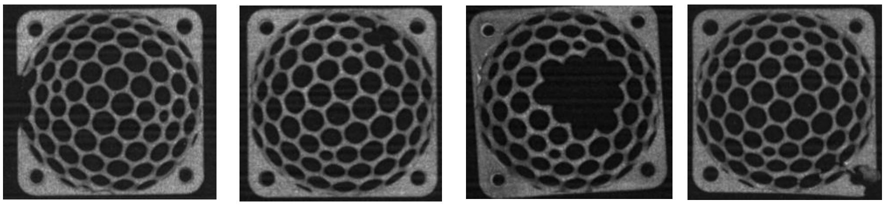


Figure 1

Two main methodologies have been tried. In our initial trial, we employed a Principal Component Analysis (PCA) to analyse data and generate a new dimensionally lower group of variables that would have been used to build a T^2 control chart. Given the poor results of this first method, we followed our second option which relies on an experimental approach.

The objective of this approach was to identify a concise subset of variables through a trial-and-error analysis. The process begins by hypothesizing which variables may significantly impact the process under investigation. These hypotheses were then tested by constructing individual control charts for each variable.

Based on the observations of those, the variables that demonstrated meaningful results were selected to create the final subset of variables.

Once this cluster was created, it would have been used to build a multivariate control chart. However, our findings led us to choose just one variable that alone was able to detect most of the problems found in the images. It is worth mentioning that the proposed method is highly effective for the types of defects that are given, therefore, to get a good result we had to sacrifice the flexibility of our design tool.

2. Assumptions

Here we will present the assumptions that outline the subsequent results. In particular, we have divided these assumptions into conditions on the images to be analysed and on how the characteristics of these images are distributed statistically.

2.1 Assumptions about Images and Pieces:

At the MADE competence centre, during the data gathering, the team representative decided to take images composed of batches of 4-6 components.

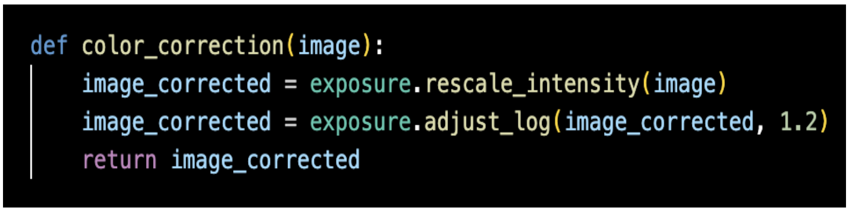
This choice made the picture analysis more challenging. In fact, it turned out, the pictures collected displayed lighting problems. In particular, due to their distance from the light sources, the more external pieces in the image presented shadows on their edges, compromising the correct identification of those.

Additionally, a perspective issue arose. Some objects were both too close to the camera lens and at the same time too far from the focal point of the frame, jeopardising the right computation of these pieces' dimensions. Naturally, this problem increases proportionally with the number of pieces per tray.

Despite our efforts to solve these problems, we concluded it would have been wiser to change the sampling strategy from the beginning. Unfortunately, this choice was not an option, therefore we chose to manually eliminate some of the pieces that the image processing was not capable to fix. After different attempts, we decided to exclude 11 out of 30 non-defecting components from our training set.

Furthermore, this approach introduces a higher workload due to the needs for analysing defects using a single piece. To get this, we initiated the process by segmenting the pieces, determining the centroid of each, and subsequently cropping a specific area around them.

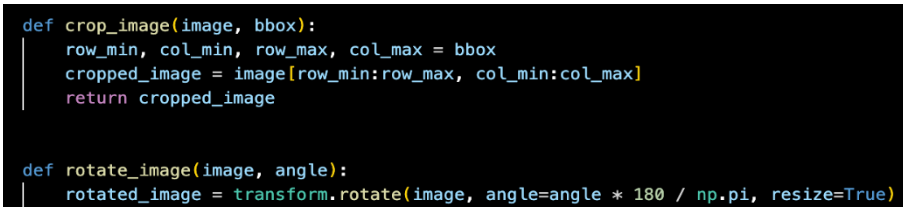
The image specifications were then 'adjusted' so that further analyses were possible. In particular, we first applied a color 'correction' on the intensity of pixels (rescaling the contrast to the intensity of the pixels) and then performed a logarithmic gamma adjustment. (Logarithmic gamma correction).



```
def color_correction(image):
    image_corrected = exposure.rescale_intensity(image)
    image_corrected = exposure.adjust_log(image_corrected, 1.2)
    return image_corrected
```

Figure 2

The images were then recorded (rotated and cropped) in order to make a standardisation of the data needed by the algorithm to do proper analyses.



```
def crop_image(image, bbox):
    row_min, col_min, row_max, col_max = bbox
    cropped_image = image[row_min:row_max, col_min:col_max]
    return cropped_image


def rotate_image(image, angle):
    rotated_image = transform.rotate(image, angle=angle * 180 / np.pi, resize=True)
```

Figure 3

2.2 Statistical Assumptions:

One problem that was encountered concerned the check for iid hypothesis. In fact, the time series of the data was unknown, thus making the randomness check infeasible. Indeed, to check the independence of the data, we first applied the run test, whereby changing the order of the data provided different results. This analysis was then further developed by looking at the correlation between data through the study of PACF and ACF. Again, manipulating the order of the data provided different results.

Due to these outcomes and the impossibility of deriving the time series, we assumed the data were randomly distributed.

Concerning the normality hypothesis, we applied the Shapiro Wilk test to variables that could be suitable for describing variations in image characteristics. In particular, the test was applied to some key variables: perimeter filled, area convex, perimeter, extent, area filled, area circle, perimeter circle, extent circle area edge, perimeter edge, extent edge.

3. Proposed method

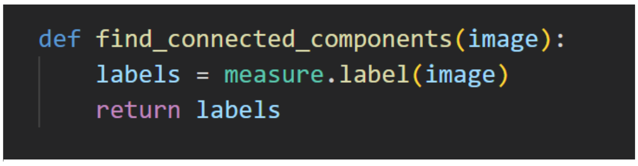
3.1 Method A

The first methodology employed is a Principal Component Analysis (PCA) with the purpose of using a lower number of variables in order to develop a Hotelling T^2 control chart. To accurately explain the variance, we initially used 18 principal components (PCs's variance almost 95%). However, as we progressed with the analysis, we observed that the PC values remained constant, and the results of the chart with the first 18 PC is rather poor, as all pieces were in control according to the chart. Also changing the number of PC was not giving any result. The ineffectiveness of the analysis could potentially be attributed to the camera used to make the pictures which provided an overall poor quality of the dataset used, which hindered our ability to utilise this methodology effectiveness.

3.2 Method B

Our second methodology starts by binarizing the images using the Otsu's method. This enabled us to separate the pieces inside the images from their background, therefore dividing the pixels representing the pieces from the ones that do not.

After doing so, a function that takes as input a binary image and assigns labels to the connected components was applied.



```
def find_connected_components(image):
    labels = measure.label(image)
    return labels
```

Figure 4

A connected component is a set of pixels connected between each other based on a certain principle. The criterion used is that neighbouring pixels should have the same value (either black or white) and be directly adjacent whether horizontally or vertically. In our case, the connected components refer to the white pixels which represent the manufactured product.

By applying labels to those, it was possible to segment and identify the objects or regions in the binary image which represents the pieces under analysis. Those labels were then used as input in the regionprops() function, which provided properties on the connected components.

From those properties, we decided to use for our analysis just the subset of variables we considered more useful for detecting errors: Area, Perimeter and Extent.

To get other variables for better detecting defects, it was needed to do 2 different further actions on the binarized images.

On one hand, we decided to apply a mask on the image in order to split it between what is inside the circle and what is outside the circle (*Figure 5*). The reasoning behind this choice was to use two different variables to detect the two main cluster of defects: Missing material along the border and broken/missing strut, respectively with Perimeter Edge, and Area Circle.

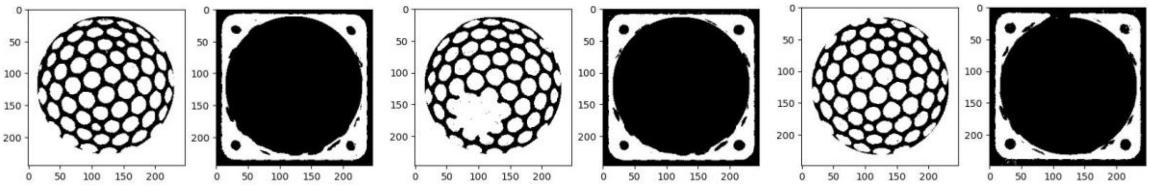
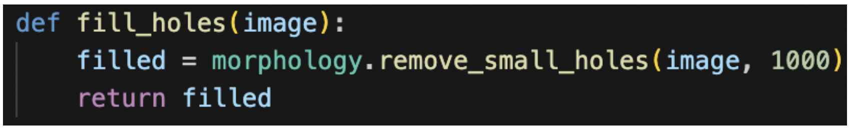


Figure 5

On the other hand, we have analysed the maximum area of the holes within good pieces. Given that threshold, we used a function to fill all the holes characterised by an area below that value.



```
def fill_holes(image):
    filled = morphology.remove_small_holes(image, 1000)
    return filled
```

Figure 6

The output of the function is illustrated in the images below, which show the comparison between the binarized image before and after the function is applied.

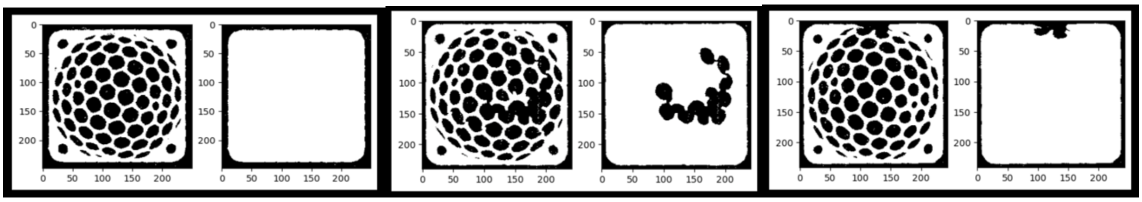


Figure 7

From such images two variables were chosen for our detection: “Area filled” which provided the number of white pixels in the image (the good pieces had higher values); “Perimeter filled” which outlines the perimeter of these white objects (the bad pieces had higher values).

The identification of all these variables was done with the purpose of carrying out a Multivariate Control Chart. Hence, we had first to check all the statistics assumptions we have previously mentioned and then develop univariate control charts on all variables (these assumptions are needed to develop a control chart that works effectively). This univariate analysis was done to understand which were the characteristics that best detect the broken products.

From now on, we will present our results just basing the analysis on the one variable used for the final proposed method: PerimeterFilled which alone outlines more broken products than those detected by all the other variables together. Therefore, our final methodology relies on a single variable Univariate Control Chart Analysis.

3.4.2 Check Normality Assumption

Starting from the design phase, the first step is to check the assumption of normality distribution for the variables to use in developing the control chart. To do so, we used the Shapiro-Wilk test which is a statistical test used to assess the normality of a dataset. Applying this test on PerimeterFilled, the p-value computed was 0.637 as shown in *Figure 8*, higher than α (α=0.05). Therefore, the data analysed can be assumed as normally distributed.

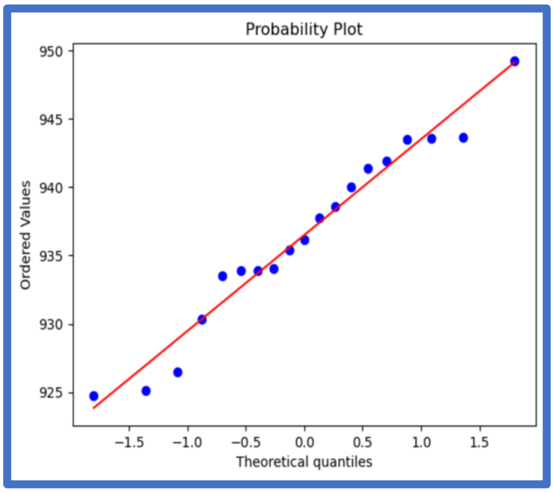


Figure 8
Perimeter Filled p-value of the Shapiro-Wilk test: 0.637.

3.4.3 Development of control charts

The first step to develop the control chart was to separate the dataset in training and testing data. Since we already know which pieces (table of defective/non defective pieces) were in control or out of control, to design the Control Chart the 19 “in control” pieces were selected in the training set.

Given the assumptions highlighted in the correspondent section, we are going to show the I-MR control chart both design and the test phase.

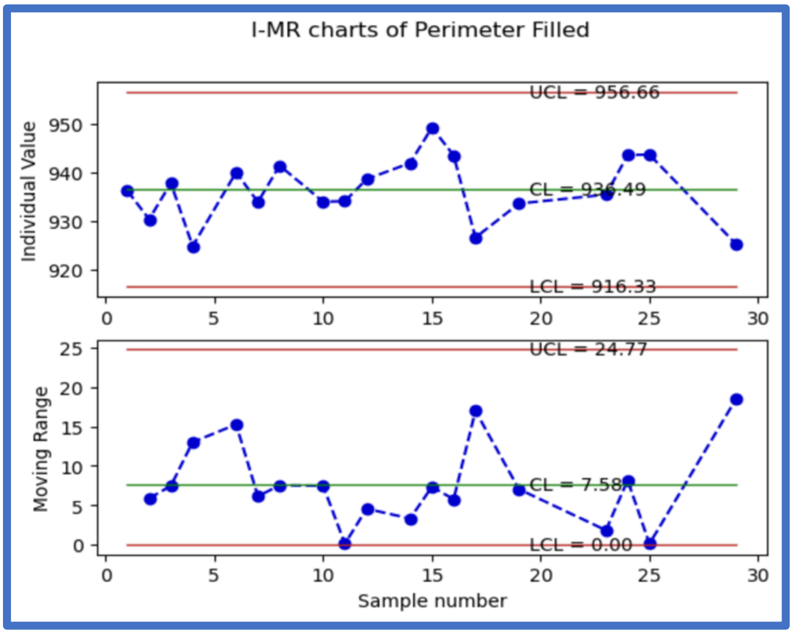
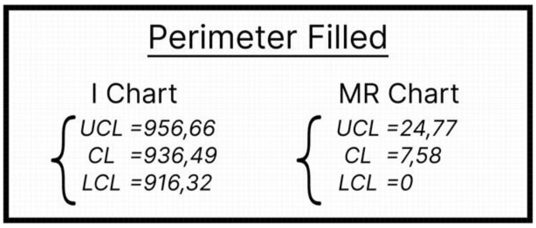


Figure 9

Following the procedure to develop a Shewhart control chart, the function employed to generate it considered a K = 3. Taking in considerations limits calculated with this value of k, no alarms are present which means there is no occurrence of first type error. This resulted in using all the available data to design the control chart resulting in computing the values of the control limits as follows:



Perimeter Filled

I Chart

UCL =956,66
CL =936,49
LCL =916,32

MR Chart

UCL =24,77
CL =7,58
LCL =0

Figure 10

4. Results

For our method validation we used the dataset provided by the professor which contains a total of 66 images of both in control and out of control pieces. From the 66 images, we discarded 8 of them since these are just a picture of an empty tray.

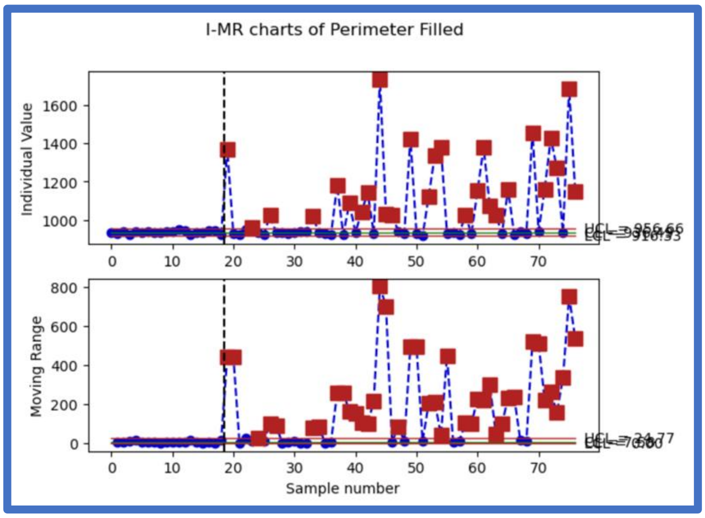


Figure 11

As we can see, the two charts are composed by a line dividing the 19 data used for the design phase and the other 58 used in the testing phase. For what concerns the data in the testing phase, it is possible to notice that the I control chart returns 27 alarms out of 58, thus returning all the pieces that presented a defection with the exception of one, the number 38.

The aforementioned results can be considered satisfactory according to three main reasons:

1. There are no false alarms in the test phase;
2. All structural errors are detected (the missing struts and the missing material along the border);
3. The second type error is low ($\beta = \frac{1}{28}$) thus, it guarantees a strong power of the test (power = 96%).

In the light of above, the primary challenge encountered relates to the identification of non-structural errors. In *Figure 12*, it can be observed that these errors are superficial in nature, resulting in image processing outcomes that closely resemble those of good images. To effectively detect such defects, the incorporation of a different camera capable of detecting better the depth on the surface of the object is needed.

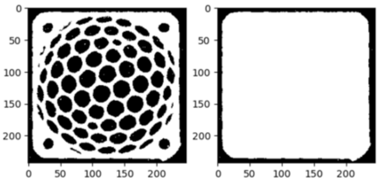


Figure 12

On the other hand, looking at the MR control chart there is a higher number of out of control (40). This 40 out of control can be divided into:
- Real broken piece;
- Out of control detected right after the broken piece.

This second event happens since the Moving Average is computed as $MR_t = X_t - X_{t-1}$. Therefore, the high difference in value with the sample immediately before (which is a true out of control), lead the value of the MR to assume an out-of-control value even if the sample is in control. Furthermore, the fourth sample is not detected in the MR control chart even if is out of control. In fact, the difference in value between the 3rd and the 4th sample falls inside the control limits.
Based on the foregoing, the analysis of the MR control chart needs to be deployed considering its limits in detecting errors.

In conclusion, the results obtained through the implementation of the proposed method far exceed our expectations and demonstrate exceptional levels of satisfaction. On the other hand, it is worth mentioning that even though our current process is capable of detecting errors, it is not able to state which specific type of error occurred (missing strut or error along the corner). Consequently, when considering a production line where errors need to be attributed to a particular cause for effective resolution, this deficiency may hamper both error prevention and targeted process improvement.

5. REFERENCES

Montgomery, D. C., & Runger, G. C. (2011). Applied statistics and probability for engineers. Wiley.

Alwan, L. C. (2000). Statistical Process Analysis. McGraw-Hill/Irwin.

Tam, A. (2021, October 27). Face Recognition using Principal Component Analysis. Machine Learning Mastery. https://machinelearningmastery.com/face-recognition-using-principal-component-analysis/