technische universität dortmund

fakultät statistik

Industrial Data Science 2 Report

Improvement of the automated classification of plug-in connections

Alp Yalcin

Enrolment Number: 230888

Course of Studies: MSc. Data Science

Submission Date:

19.08.2022

Academic Advisors:

Prof. Dr. Markus Pauly

Prof. Dr. Erich Schubert

Prof. Dr.-Ing. Jochen Deuse

Prof. Dr. Jens Teubner

Contents

List of abbreviations

e. g. for example

i. e. id est

MSE mean squared error

MAE mean absolute error

RMSE root mean absolute error

nOK nok ok

List of Figures

List of Tables

1 Presentation of the Problem

Assembly process is one of the most significant part of the in manufacturing of engines. With the improvement of technology, the assembly process is held by automated machines more instead of human power which makes the process faster, more convenient, and more practical way. However, it is crucial to have correct model for the specific automation solution.

In the BMW engine assembly process, control units, actuators, and sensors installed on the engine are connected to various cable strands. These connections are proceeded with by cable plug connections to speed up to assembly process. In order to decide whether the plug has been plugged correctly or not, there is an automatic camera station that analyses the individual plug connections using reference points and reports the result at the end of the assembly line.

On an irregular basis, the camera station reports an error although the connector was plugged in correctly, which is called pseudo-nOK to the connector being subjected to an unnecessary visual inspection.

Numerous plug connections are classified as not OK, which are OK by the camera station. Even tenuous differences in the illumination or reflection of the material surface can trigger to the connection not being detected correctly.

Additionally, the other problem is, that some plug connections cannot be defined by the camera station whether these connections are OK or nOK due to some cables or other assembly parts are obstructing the camera station to proper imaging. These are called undefined values in this report.

2 Data analysis and Classsification Methods

2.1 Classification of the Problem in the Task Context

Supervised learning algorithms are, roughly speaking, learning algorithms that experience a dataset including features associated with a label or target. A supervised learning algorithm learn to classify based on the feature properties.
(Goodfellow et al., 2016, p. 125)

In the method which is applied, binary class classification is used for this problem. As it explained before, automatic camera station analyses the binary case of the plugs whether the connector plugged correctly or not. Therefore, the model approach is focused on this binary classification method.

For tasks such as classification, the accuracy of the model is measured. Accuracy is the proportion of samples for which the model generates the correct output. However, sometimes accuracy is not the best way to characterize the performance.

The other way is using precision and recall measures. Precision is the fraction of the findings reported by the model that were correct, while recall is the fraction of true events that were detected. Therefore, precision is calculated as true positives divided by the sum of true positive values and false positive values. On the other hand, recall is calculated as true positives divided by the sum of true positives and false negatives.
(Goodfellow et al., 2016, p. 125)

Confusion matrix is used to evaluate and compare the scores of various methods which are tested and showed on the last section of the report. Main purpose to solve this problem is increasing the classification accuracy of the camera station on connector plugs.

Figure 1: Image Classes

In the plug-in connections problem, the automatic camera station decides whether it is plugged correctly or not from the red dashed rectangle line part which is shown in the pictures. If the bottom of the half circle lattice(cable) has three small marks(lines), then this plug connector is not plugged correctly. This bottom side should be as smooth as the other parts to plug properly. As shown in the pictures, sometimes the camera station cannot clearly take a picture of the connections since some pipes or other parts are blocking the bottom of the rectangle part, which is the major sign to determine whether the connection is plugged correctly or not. Then, it becomes harder to determine the plug connection; it is called Undefined observations.

2.2 Explorative Data Analysis

At the beginning of the project, 899 observation photos from plug connections are provided with two different folders named OK and not OK. Dataset of the photos has been collected in the three months while producing more than 100.000 motors. The number of 399 values is classified as not OK and belongs to the related folder, and the rest 500 values are located in the OK folder and classified as correctly plugged. The default camera station classifier correctly classifies all observations in the OK folder. Therefore, all 500 observations belonging to the correctly classified folder are predicted correctly as good to go. However, this classifier wrongly predicted some of the OK observations as not OK. Briefly, the classifier reports errors in some observations as pseudo-NOK even if it is correctly plugged.

As the first step of data analysis, all 899 observations from both folders are manually checked, and wrongly classified observations are detected. The real number of not OK values was given as information from the data provider before with the number of 49 values. As a result of manually checking, 345 observations are recorded as pseudo nOK and 48 observations are correctly classified as not OK values detected. 6 values are classified as undefined since photos of these observations are indistinguishable whether it is plugged correctly or not. Therefore 48 observations out of 49 real not OK values are detected, and the only one value which cannot be detected is one of those 6 values which cannot be distinguishable.

		Predicted Class		
		Positive	**Negative**	
Actual Class	**Positive**	TP (=True Positive) 500	FN (=False Negative) 350	Sensitivity 0.5882
	Negative	FP (=False Positive) 0	TN (=True Negative) 49	Specificity 1.0
		Precision 1.0	Negative Predictive Value 0.1228	Accuracy 0.6106

Figure 2: Confusion matrix

As a result of confusion matrix, true positive values are 500, false positive values are 0, therefore Precision score of the model is 1.0 since there is no any not OK value which is classified as OK. Additionally, the false negative values are 350, and the number of true negative values are 49. Thus, negative predictive value score (NPV) is 0.112 which is quite low score in this case. Also, sensitivity score is 0.58 and specifity score is 1.0. Accuracy score of the model yields 0.611 which has to be improved significantly.

2.3 Justification for Selection of a Method

In this project, thresholding methods are applied to find a best solution for the problem.

2.3.1 Thresholding Methods

In this part of the report, general thresholding methods which are simple thresholding and Otsu's method is explained. Also adaptive thresholding is clarified.

Simple Thresholding

Simple thresholding is used when parameters are manually supplied to segment the image. Usually, simple thresholding works well in controlled lighting circumstances where high contrast between the foreground and background of the image is ensured. Application of simple thresholding requires manual interposition. Threshold value T has to be specified manually, pixel densities below T are set to 255 and above T are set to 0, or vice versa. Rosebrock (b)

After importing the related packages, images have to be preprocessed by both converting them to grayscale and implementing Gaussian blur. Gaussian blur application removes some of the high-frequency edges in the image that is not related to the application and helps obtain more pure segmentation.
Rosebrock (b)

After the preprocessing, thresholded image is computed with related function. In simple thresholding method function, arguments of the function helps to choose image which was blurred before, also helps to supply T threshold value manually, output value applied and option of preference of thresholding method. In simple thresholding, binary method or inverse binary method can be chosen from the function.
Rosebrock (b)

Last not but not least, also image masking can be applied in simple thresholding with using the thresholded image as a mask. Image masking only consults the pixels of the original image where the mask is greater than zero.

Different lightning situations might affect the performance of the simple thresholding method, since thresholding value is provided manually which might be hard to find the optimal value. There is no assurance that threshold value T works well from one image to another in the presence of lighting changes. Particularly, for creating a dynamic system and several lightning situations this inflexibility can be real problem for real life applications such as plug in connections in our case.
Rosebrock (b)

Other general thresholding method called Otsu's method or adaptive thresholding can be more helpful for better results.

Otsu's Method

In Otsu's method, the optimal value of threshold T is computed automatically. In real-world conditions where the prior information of the lighting situations is unknown, Otsu's method is a better solution than simple thresholding.
Rosebrock (b)

Otsu's method assumes that our image includes two classes of pixels: the background and the foreground. The grayscale histogram of pixel densities of the image is bimodal. Therefore histogram has two peaks. The optimal threshold value T is calculated with minimum variance between the background and foreground peaks. On the other hand, Otsu's method tries to separate the peaks from the histogram since it does not have prior

information on what pixels belong to the foreground and background.
Rosebrock (b)

Otsu's method-related function can use blurred images again, just like in the simple threshold method. The argument which has to specify a threshold value can simply be set to any value since it will be calculated automatically anyway. The type of thresholding that is performed is calculated with the two flags with OR statement to correspond to Otsu's thresholding method.
Rosebrock (b)

The result is optimal if the best possible option is to split the foreground and background, assuming the bimodal distribution of grayscale pixel values. However, if the grayscale image does not follow a bimodal distribution, Otsu's method may not give the optimal result. In that case, adaptive thresholding can be a better option for using thresholding on images.
Rosebrock (b)

Adaptive Thresholding

In some situations, variations in lighting circumstances, shadowing, single thresholding T value may not perform enough on different segments. If the lighting circumstances are not uniform, global thresholding methods such as simple thresholding and Otsu's thresholding is not provided the best solution. In that case, adaptive thresholding can be a more efficient method.
Rosebrock (a)

Adaptive thresholding consults a small set of neighboring pixels, calculates T for the local region, and applies the segmentation. The adaptive threshold can be helpful to obtain better segmentation than global methods and also a good option to avoid the time consumption of complex processes such as CNN or the U-net segmentation network.
Rosebrock (a)

As mentioned before, in adaptive thresholding, the aim is to statistically analyze the pixel density values in the neighborhood of a given pixel, p. Small regions of an image tend to have approximately uniform enlightenment. This indicates that local areas of an image have similar lighting compared to the whole image. Therefore, the selection of the size of the pixel for local thresholding plays a crucial part in the performance. The neighborhood has to be large enough to cover adequate pixels to avoid irrelevant T values. On the other hand, the neighborhood also has to be small enough to have uniform illumination in local regions.
Rosebrock (a)

The arithmetic or Gaussian mean is used to calculate the pixel intensities in each

region. In arithmetic mean, each pixel in the neighborhood contributes equally to the calculation of T. In Gaussian mean, pixel values farther away from the region's center contribute less to the overall computation of T.
Rosebrock (a)

In adaptive thresholding, related functions and arguments are similar to the other methods explained above. Only, in the third argument where the thresholding method is indicated, there are two selections on which calculation method is used in adaptive thresholding ie. Arithmetic mean or gaussian mean. Additionally, in the fifth argument, pixel neighborhood size is implemented, and in the last argument, a constant value used to tune the thresholding is selected.
Rosebrock (a)

To sum up, adaptive thresholding produces better results than global thresholding methods, especially in non-uniform illumination situations, however, it is more computationally expensive than global thresholding methods.

2.3.2 Regression Metrics

In this report, three different model performance matrices are examined. These performance evaluators are mean squared error (MSE), root mean squared error (RMSE) and mean absolute error (MAE).

Mean Squared Error(MSE)

Mean squared error calculates the average squared difference between the observed value of an image coordinate and value of another image on the same coordinate. The lower MSE score means the better result for the similarity of the images.

Root Mean Squared Error(RMSE)

Root mean squared error metric calculates the square root of the average squared difference between the observed value of an image coordinate and value of another image on the same coordinate. The idea of the RMSE metric is quite similar to MSE metric, only while calculating the score, the square root has to be taken. For binary images RMSE ranges between 0 and 1. Again the lower RMSE score means the better result for the similarity of the images.

Mean Absolute Error(MAE)

The MAE score is calculated as the average of the absolute error values. The difference between the observed value of an image coordinate and value of another image on the same coordinate could be positive or negative and will change to positive after the MAE score is applied.

1	0	1
1	0	0
1	0	0

Image A - 3x3 Pixel

0	0	1
0	0	0
0	0	0

Image B - 3x3 Pixel

$$MSE = \frac{(1-0)^2+(0-0)^2+\cdots}{9} = \frac{1}{3} \; ; \; RMSE = \frac{1}{\sqrt{3}}$$

Figure 3: Regression metrics

2.3.3 Image Methods

While training the model, mean image method and random image method are used and performance of these two methods are compared.

Mean Image

In mean image method, specific amount of images is used for having mean classified images to compare with the other observations. For instance, in plug in connections case, 8 OK images and 8 Not OK images is taken for training of the model. The average of all color pixels of each 8 images is taken and at the end, with the mean image method, 1 OK and 1 nOK averaged trained images are procured and used to compare and classify the other observations.

Random Image

In random image method, the process is simpler than the mean image method. Only 1 OK and 1 not OK images is taken as a training set (or reference set), and only these 2 observations is used to make classification of the other observations. Mean image method is more complex and accurate than random image method since more observations is used to train the model.

2.4 Description of a ML method for the Application

In the ”Improvement of the Automated Classification Of Plug-in Connections” project, various methods is applied for attaining the optimal result. Mean image and Random image methods is implemented. Also, global thresholding methods and adaptive thresholding method is used. For the performance metrics, RMSE, MSE and MAE regression metric measures are implemented. Both performance accuracy and runtime of these implementations are considered.

In mean image model, 8 observations for each class has been trained and the average pixel colours of these observations is taken and one averaged image is procured for comparison with performance metrics. In random image model, simply 1 random OK and not OK images are taken as training set and these images are used to comparison of regression metric scores for the rest of the observations for classification.

For achieving more consistent and correct results, image sizes are decreased from 150x150 pixels to 24x24 pixels since only this small part of the image is important for the classification and decreasing point of interest of the images helps to get more accurate regression metric results and also it accelerate the learning and algorithm processes.

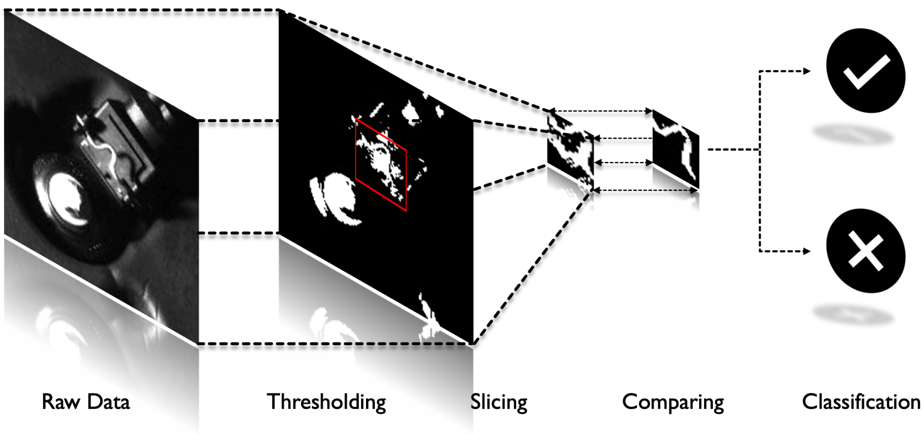

Figure 4: Classification

Regression metrics are used to compare the differences between the training images and the test images. Each pixels of the trained image, for both mean image model and random image model, is compared with the image which will be classified. Color pixel differences are taken account with regression metrics and the test images are classified with these scores. If the differences between the reference OK image (trained image) and the image which is going to be classified is smaller than the not OK image(trained image) and the image which is going to be classified, then this observation is classified as OK image. Otherwise, it will be classified as not OK image. This classification method is quite simple but significantly effective.

2.5 Presentation of Validation Results

Table 1: Comparison of various Results

Method	Threshold	Metric	TP/P	TN/N	NPV	Sensitivity	Accuracy	Run.
Mean	Global	RMSE	752/788	17/111	0.133	0.889	0.855	3.4s
Mean	Global	MSE	304/304	48/595	0.081	0.357	0.392	4.6s
Mean	Global	MAE	304/304	48/595	0.081	0.357	0.392	3.8s
Mean	Adaptive	RMSE	849/851	46/48	0.958	0.998	0.996	4.0s
Mean	Adaptive	MSE	818/820	46/79	0.582	0.961	0.961	4.6s
Mean	Adaptive	MAE	841/846	48/53	0.901	0.994	0.989	4.5s
Random	Global	RMSE	733/771	10/128	0.078	0.860	0.827	3.8s
Random	Global	MSE	412/412	48/487	0.099	0.484	0.512	4.8s
Random	Global	MAE	412/412	48/487	0.099	0.484	0.512	3.8s
Random	Adaptive	RMSE	797/805	40/94	0.426	0.937	0.931	3.7s
Random	Adaptive	MSE	790/795	43/104	0.413	0.928	0.927	4.2s
Random	Adaptive	MAE	819/824	43/75	0.573	0.962	0.959	3.7s

As it explained above, different kind of methods is used to find the optimal solution for the best classification algorithm. In this part of the report, all method results will be examined in detail.

For the mean image method with global thresholding, it gives the best runtime score which is 3.4s with RMSE metric. Although the true positive values are perfect for the MSE MAE metrics with 304/304 observations, mean image with global thresholding is not able to find the negative observations correctly. The TN / N scores are so low for all three metrics. RMSE metric sensitivity and accuracy scores are the highest one with 0.899 and 0.855, respectively, compared to other metrics for the mean image method with global thresholding. The other metric scores are quite low in between the scores of 0.35 and 0.40.

For the mean image method with adaptive thresholding, RMSE metric scores give the best overall scores for the project. This method with RMSE metric has 0.997 true positive rate, 0.959 true negative value scores. It has 0.958 NPV value and it has the best adaptive thresholding method runtime score with 4.0 seconds. Last but not least, mean image method with adaptive thresholding with RMSE metric has the best sensitivity and accuracy scores with 0.998 and 0.996, respectively.

When the global thresholding and adaptive thresholding methods are compared, it can be seen that runtime scores for adaptive thresholding runtime is higher than global thresholding metric scores. However, in RMSE metric of adaptive method, the runtime is also quite fast which is one of the reasons why we choose this method over the other ones.

For the random image method with global thresholding, it can be seen that all three metric scores are average and these scores do not give the optimal results. Again, in MSE and MAE metrics, it can be seen that true positive rates are quite good but NPV scores are not acceptable. In random image global thresholding method, RMSE metric give the best sensitivity and accuracy scores with 0.86 and 0.83.

For the random image method with adaptive thresholding, sensitivity and accuracy scores are better than the global thresholding. This time, MAE metric gives the best sensitivity and accuracy scores with 0.962 and 0.959, respectively.

To sum up, it can be seen from the comparison of various methods, mean image method gives better solution than random image method. Additionally, adaptive thresholding gives better results than global thresholding, however runtime should be considered too in order to have reasonably fast solution. Therefore it is more logical to consider mean image method with adaptive thresholding algorithm.

3 Conclusion/Outlook/Critical Appraisal

In this project, the Automated Classification Of Plug-in Connections was examined, and possible solutions to improve these connections were studied. Firstly, the image dataset provided before was classified manually, and the built-in classification algorithm of the camera station was viewed. Pictures classified as correctly plugged were classified correctly; however, most of the pictures classified incorrectly as not plugged correctly had to be improved.

Mean and random image methods were applied with global and adaptive thresholding to compare the results and find the optimum solution. One of the team's main aims was to have an optimal solution without complex, time-consuming machine learning / deep learning methods and to find the most practical solution with considerably fast running time.

In the mean image method, 8 images were used as training data for each class (nOK and OK), and the mean image method was trained with 16 images from both classes. For the random image method, one random picture for each class was taken as a reference class, and all other images were classed as regards these reference single images.

POV(point of view) of the image pixels was decreased to 24*24 from the original picture size of 150*150 to improve the prediction by only focusing on the interesting areas of the images, i.e. only the cable connection area.

Different kinds of metrics performance measures was used to have the best solution and all of these results was examined individually for both methods. As a final decision, mean image with adaptive thresholding method with RMSE performance metric was chosen as a best model with 0.958 NPV score, 0.998 Sensitivity score and 0.996 Accuracy score for this specific problem. Also, runtime of the algorithm was considerably fast with 4.0 seconds for 899 image observations.

Mean images adaptive thresholding method performed significantly adequate for the dataset which was provided at the beginning of the project, however, when classifying another cable plugs or other connections at different motor positions, the algorithm might fail without more training data. Additionally, in this project, similarity between the images is measured to classify whether they are plugged correctly or not. Therefore, noisy training data might be problematic to handle for the current model. For further improvements, it can be studied how to handle noisy training data observations and how to adapt this specific model to the other plugs or connection at different model positions to have more useful and convenient solution.

References

Goodfellow, I., Bengio, Y., und Courville, A. (2016). *Deep learning.* MIT press.

Rosebrock, A. Adaptive Thresholding with OpenCV (cv2.adaptiveThreshold).

Rosebrock, A. OpenCV Thresholding (cv2.threshold).