International Journal of Engineering & Technology, 10 (2) (2021) 148-154

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Using Haar-like Features and SVM Classifier for Quality Assurance in a Surgical Mask Production Line

Laszlo Marak[1]*

[1]*J. Selye University, Bratislavská cesta 3322, 945 01, Komárno, Slovakia*
**marakl@ujs.sk*

Abstract

With the recent increase for demand of surgical masks, the design and development of mask production lines has become an ever pressing issue. These production lines produce low cost high quantity products. As there are errors during the production, it is important to be able to detect invalid masks to assure that the produced masks are of consistent quality. Manual quality assurance using human operators is an error prone and a costly solution. In this article we describe an image classification method, which is using a low-cost Commercial Camera System and relies on Haar-like features combined with Maximum Relevance, Minimum Redundancy feature selection to detect the invalid masks at the end of the production process. The classification method consists of Preprocessing, Feature Selection and SVM Training. We have tested the method on a database of 150 000 images and it provides a high accuracy method which we use in the Production Line.

Keywords: *Support Vector Machine, Haar-like Features, Industrial Image Processing, Quality Assurance, Machine Learning Applications*

1. Introduction

High throughput production lines pose a significant challenge to Quality Assurance Services. Detecting of invalid products on the production line can be complicated as multiple products can exit the production line each second. It has been a long established tradition to automate Quality Assurance tasks using CMOS Cameras. Some of these methods rely on semantic algorithms, such as morphology or segmentation. While these methods provide an excellent solution to many problems, they have certain limitations. For example, each case needs specific consideration, and therefore there are no off-the-shelf solutions which could provide the factories with a flexible tool. Most of image processing libraries either provide base operators: OpenCV [2], ITK [9] and IPP [15]; or provide a turn-key solution to a specific problem, such as FaceTrainer in OpenCV.

Machine Learning can be an alternative to these methods, as it can provide the factories with the possibility of adjusting the algorithms, without modifying the source code of the programs. Machine Learning also provides a possibility to create algorithms for cases where there are no precise semantic requirements, such as the exact position of the masks and the folding. As Machine Learning is mostly relying on annotations for the training, we can accommodate all the explicit and implicit requirements provided by the human experts. Authors of [11] have recently compiled an overview of Machine Learning algorithms for Industrial Image Processing.

In this article we describe a novel application which relies on a combination of histogram equalization, Haar-like feature extraction, MRMR feature selection and SVM classification to train a classifier which is able to detect invalid products in real time using a low cost commercial PC. We have tested our method on a real-world annotated database which contained 150 000+ images captured during the production process.

2. Prior Approaches

There have been numerous applications for SVM in Industrial Image Processing. Chittilappilly et al. applied SVM with non-local denoising method, color histrogram and geometric moment features to build classifiers for material defect detection [4].

Wu and Lu used SVM for detecting surface errors on the products' packaging [18]. They have experimented with different statistical features, such as contrast, and used the classifier to distinguish between faulty and acceptable products. Their setup is similar to our approach, in that they are mounting a camera on the production line to examine the product in the final production step.

[1]The research have been supported and the images provided by Cloud Network Technology Kft. 2900 Komárom, Bánki Donát utca 1., subsidiary of the Foxconn Technology Group

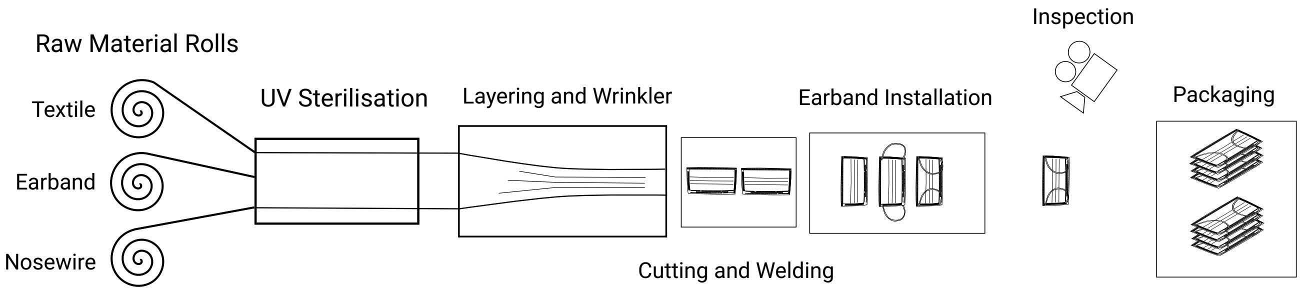

Figure 1: The Schematic of the Mask Production Line

Authors Yang et al. present yet another approach in which they compare SVM with CNN Deep Learning [19]. Their method learns the features during the training process, which thus explicitly avoids manual feature specification or feature selection. Their method performs better than SVM classifier, albeit they have less control over the learning process.
Garten et al. examine image processing possibilities for industrial applications [7]. With the advent of microelectronics such as microcontrollers, embedded General Purpose GPUs (GPGPU) and especially Field Programmable Gate Arrays (FPGA), there is a growing interest in using image processing methods as parts of the industrial production lines. The article considers several datasets and successfully applies SVM for grain and material identification.
In an article [20], authors Zhao et al. experiment with SVM for the detection of faulty road surfaces. Their problem is different from ours in that they are working with unconstrained images as the images have to be recorded in the field under variable light conditions. They interest and main focus lies in the surface texture and color. The authors develop an eigen-vector based color feature, which they successfully apply to their dataset.
Among other promlems, SVM has been successfully applied for multispectral satellite images for identifying vegetation on different land areas [1], invasive plant detection (Avena Sterilis, [16]), classification of MRI images [3] with many other possible applications.
Aside from the machine learning algorithm itself, feature selection is an essential component of a successful classifier training. Authors [5] review several feature selection algorithms. The subject of their study is Industrial Food Quality Assurance. They review several feature selection algorithms, from linear methods, like PCA to advanced feature selection methods, like Convolutional Neural Networks. The authors find that feature specification and selection methods can heavily influence the performance of the classifier. SVM has also been used in combination with Fuzzy Logic [10].
Specifically Haar-like features have been used mostly for building detectors, such as face detectors [17], or detection and classification of vehicles [12]. These applications are usually common in that they would like a classifier which operates under a time constraint. While our problem also represents a time constraint, we only need to process a single image, therefore we can evaluate the complete set of selected features and use them in conjunction with a machine learning algorithm instead of a cascading classifier. We could also use alternative features like Local Binary Patterns as suggested by [8], but we have found in our study the Haar-like features performed well.

3. Description of the Production Line

The schematic of the production line can be seen on Figure 1. The production starts with the sterilization station, after which the continuous textile, rubber band and wire are forwarded to the pressing station. The pressing station combines the layer of the mask, creates a fold in the middle and installs the wire on the nose part. In the next production step the machine cuts the masks into individual rectangles and welds the sides to create a single mouth piece. In the fourth step of the production the machine installs the earbands. The earbands consist of elastic material and are used to hold the mask on the person's face by hooking around the person's ears. The bands are first cut, then welded and finally folded to their final position. The last step of the production is the packaging, which stacks 10 masks and inserts them into a plastic wrapper, which is then sealed. As indicated on Figure 1., we have installed an industrial camera overlooking the masks exiting the production line between the earband station and the packaging station.

4. Statement of the Problem

As there is variance in the raw materials, and as the production line is mechanized, there are variations in the products that exit the production line. Inspecting every mask visually by a human expert is an error prone task, which slows down the production speed and increases the product's cost as there must be more human technicians present during the production. We wanted to improve the process by providing an automatic Quality Measurement method which would be able to detect invalid masks. The invalid masks can be removed using an automatic tool from the production line. The invalid masks can be either discarded or inspected manually. As the masks which pass the Quality Tests constitute the majority of the production, verifying the invalid masks decreases the amount of work needed to be performed by human experts. The aim of our work was to develop an automated methodology, which would be able to assert the Quality of the Product (each individual mask) without human interaction.

5. Description of the Method

Our method consists of four steps: *a)* preprocessing, *b)* feature extraction, *c)* feature selection and *d)* SVM training. In this section we describe each method.

Preprocessing. In the first step we normalize the image. Even though the production lines provide a relatively controlled environment, it can still be useful to equalize the images. The equalization can filter the variations of the illumination when using non-calibrated light

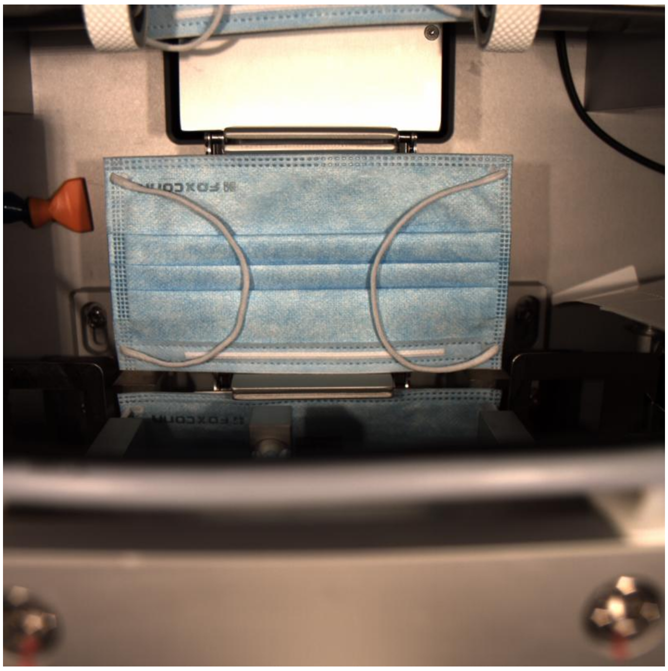

Figure 2: Sample Image of the Mask Exiting the Production Line

sources. It can also be useful for new installations, as the new installation will provide similar images to the previous stations. In our application we used the Contrast Limited Adaptive Histogram Equalization (CLAHE). This method divides the image into small blocks, or patches, then each patch is histogram equalized using the common method. In addition to the local histogram equalization, the method also considers the amount of contrast in the equalized patches. There reason for this is, that if a patch contains noise, than this noise could be amplified by the equalization. To mitigate this issue, the method collects the pixels above a certain contrast threshold and distributes them between different patches. In the final step of the method we interpolate the borders of the patches using bi-linear interpolation to account for the possible differences on the borders of the patches.
Histogram equalization has been developed for greyscale images. In many cases it is beneficial to work with greyscale images or even greyscale cameras, as greyscale sensors can be more consistent in luminosity and can be more more sensitive to light, which can result in images with lower noise. In our case, however, where illumination is not an issue, color perception can provide additional information for the classifier. We have, therefore, used CLAHE on color images by converting the images into LAB color space, normalizing the luminosity component.

Feature Extraction. In order to be able to classify the images we are using the Haar-like features in an unusual application. While mostly used for creating detectors, we have found that it can be equally suitable for classification as well. The Haar-like detectors are similar to wavelets. Each feature has a specific size and position. The detectors are divided into black and white halves, either horizontally or vertically. We use the black and white parts of the feature as masks and we subtract the sum of the pixels from the black area from the sum of the pixels in the white area. The final response of the feature is a single number (usually integer), which represents the difference in the pixel intensities between the two halves.
The Haar-like features can be normalized and calibrated. During the normalization process, we divide the response by the number of pixels. This normalization assures that the response will be bounded. This makes the Haar-like features size independent, as features of smaller area can be compared to features of bigger are without resulting in significant differences between the responses. On the other hand, having small features, such as a single black and a single white pixel, can result in a disproportionate variation of the responses. We can mitigate this effect by excluding small features. In a classification framework, we consider the class of the Haar-like feature the sign of its response. The range of the pixel differences is $-255\ldots255$ on 8bit greyscale images. We can calibrate the Haar-like features by introducing a **bias** b and a **sign** s in the classification. We have chosen the bias and the sign to maximize the accuracy of the classification. We do this by sorting the responses from the smallest to the greatest and pick the bias which results in the highest accuracy.

$$\max_{b,s} \mathrm{TFR} + \mathrm{TPR} \tag{1}$$

Feature Selection. After we have extracted and calibrated all the candidate Haar-like features and have excluded the small features, we have retained in excess of 700 000 features. The number of features depends on the smallest size of the allowed Haar-window and the size of the image. We have also excluded regions, such as image borders, which we have believed were not relevant for the classification. We had to reduce the number of the features due to the concentration of distances and also to be able to maintain a real-time performance of the trained classifier. While the original Viola-Jones algorithm is using the AdaBoost method, we have chosen a different approach. The AdaBoost method has been designed to maximize the speed of the classification. This is important for the detectors, as they have to calculate the response for many sub-images on each image. The classification, however has only one image to classify for each cycle, so we could evaluate all the selected features for each image. We have selected the Maximum Relevance Minimum Redundancy method (MRMR) of [13]. The method is based on Mutual Information (MI). Let C and D be two classifiers, the mutual information of the two classifiers for the binary case can be than written as:

$$M(C,D) \doteq P_{C>0\wedge D>0}\log\frac{P_{C>0\wedge D>0}}{P_{C>0}P_{D>0}} + P_{C>0\wedge D<0}\log\frac{P_{C>0\wedge D<0}}{P_{C>0}P_{D<0}} + P_{C<0\wedge D>0}\log\frac{P_{C<0\wedge D>0}}{P_{C<0}P_{D>0}} + P_{C<0\wedge D<0}\log\frac{P_{C<0\wedge D<0}}{P_{C<0}P_{D<0}}$$

The essence of the MI measures the dependence between the two classifiers, that is how much information (bits) can we collect about D by observing C. The purpose of MI is twofold. On one hand, we would like the selected features to be close to the annotation that is they would have high mutual information, on the other hand we would like the selected features to be independent, that have low mutual information between themselves. Let us suppose that we have $T_i,\ 0 \leq i < n_T$ training images. Let $H_j,\ 0 \leq j < n_H$ be the set of the candidate Haar-like features, so that $\forall i, j$, $H_j(T_i)$ is the response of the j-th candidate classifier to the i-th training image. Also we have an annotation A, where $A(T_i)$ represents the annotated class of the i-th image. That is $A(T_i) > 0$ if the image has been annotated as correct and $A(T_i) \leq 0$ if the image has been annotated as faulty. As a trade-off between relevance and redundancy, MRMR uses the following iterative approach for selecting the features:

a) In the k-th iteration, let $S_i,\ 0 \leq i < k$ be the set of the selected candidate features.
b) For $k = 0$, we select

$$S_0 = \max_i M(A, H_i) \tag{2}$$

c) for $k > 0$, we select the feature which is closest to the annotation and farthest away from the already selected features:

$$S_k = \max_{i \notin \{S\}} \left[M(A, H_i) - \tau \sum_{j \in \{S\}} M(H_i, H_j) \right] \tag{3}$$

with the number of selected features and τ being a parameter of the feature selection.

Classification. The nature of the training set implies that the problem is best suited for supervised learning. There are many classification algorithms in the literature as well as there are many implementations, with no clearly superior approach. We have selected the SVM classifier from [6] for the following reasons: The size of the training set is well suited for the SVM classifier. The database of 150 000 images fits on a single computer and therefore we can use a direct learning approach. The SVM classifier requires few parameters, notably the choice of the kernel, the γ kernel parameter and the μ training parameter. Finally, the trained classifier provides the number of selected Support Vectors as feedback for overfitting. This method allows us to interpret the overfitting of the training based on the proportion of the selected Support Vectors.

6. The Training Database and Training Software

We have created a database using the production line by installing a Basler acA2040-25gc 4 megapixel industrial camera surrounded by two LED panels. We have installed a camera trigger circuit after the last step of the production line. The camera is triggered each time after the mask had exited the production and before the mask has entered the packaging.
We have been recording the images of the mask during a period of several weeks on the production line and have collected an excess for 150 000 images. We have initially used Open Coding as described by [14] to annotate the images. We have developed separate codes for different faults in the production system, such as invalid texture, invalid earband, empty tray, double mask or occlusion. We have coded 42 942 images. After the coding, we have separated the images into two categories: accepted and rejected. As most of the images contained accepted products, we had much more accepted images than rejected. To mitigate this difference, we have selected the highest possible amount of rejected images, and randomly selected same amount of accepted images. This way the final training database contained 12026 annotated images. Even though, in the final classification we had only two classes (accepted and rejected), the is still a reason for Open Coding. Open Coding creates natural categories, that is classes which are both numerous and clearly identifiable. Using these classes we can approach the QA multiple ways: *a)* We can take a conservative approach, in which we reject all but the unquestionably flawless products. *b)* We can also isolate the fault types and estimate their frequency, both in the training set and during the production. This way we can exclude artificial categories, which are not represented in practice, or *c)* we can revise the training set to create a more relaxed classifier, which accepts the masks if the fault is not critical, such as wrinkling of the material.
In the feature extraction phase of the training, we have extracted 715 328 Haar-like features, both horizontal and vertical. During the selection we have excluded small features below a certain size and excluded parts of the image which were not relevant to the class.
After the feature extraction we have experimented with different number of selected features. We have to consider both overfitting, computation time and the concentration of distances in order to select the optimal number of features. It is also important to note, that the exact choice of the number of features should not influence the accuracy of the classifier. If adding or removing a few features from the selection deteriorates the accuracy than there is a risk of overfitting. In the final version we have used 40 features selected using MRMR. We have further compressed the 40 features into 25 using PCA with 99% retention rate.
For the training phase we have selected Support Vector Machine, with Radial Basis Function kernel. In the parameter search phase of the training we have used 5 fold cross validation with $10^{-5} \leq \gamma, \mu \leq 5$.

7. Results

We have tested the classifier using both the annotated database and through manual verification. The selected features are shown on Figure 3. The corresponding ROC can be seen of Figure 6. We have also presented a few rejected and accepted images on Figure 5. and Figure 4. respectively. These images have not been annotated, so where not considered during the training. The final classifier, including the bias attains 95% accuracy.

8. Conclusion

Using automated image processing for Quality Assurance in production lines is increasingly common. Machine Learning provides an excellent solution for these problems, as it can function without precise mathematical description of the problems. As it relies on Annotated

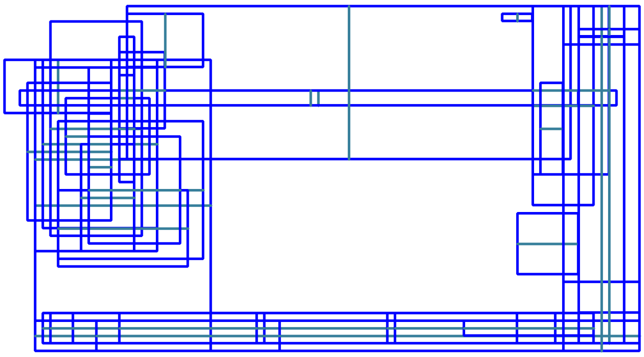

Figure 3: The Selected Haar-like Features

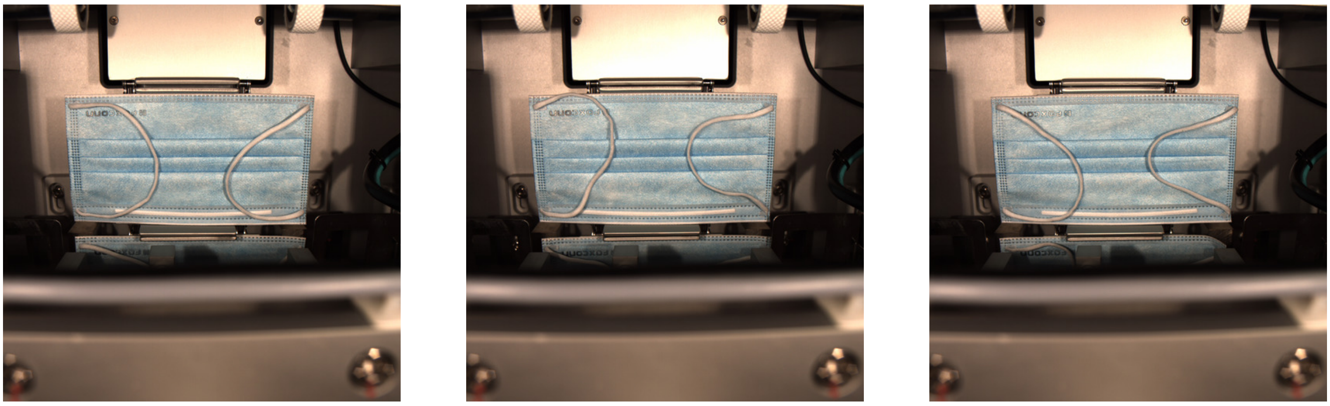

Figure 4: These images where accepted by the classifier

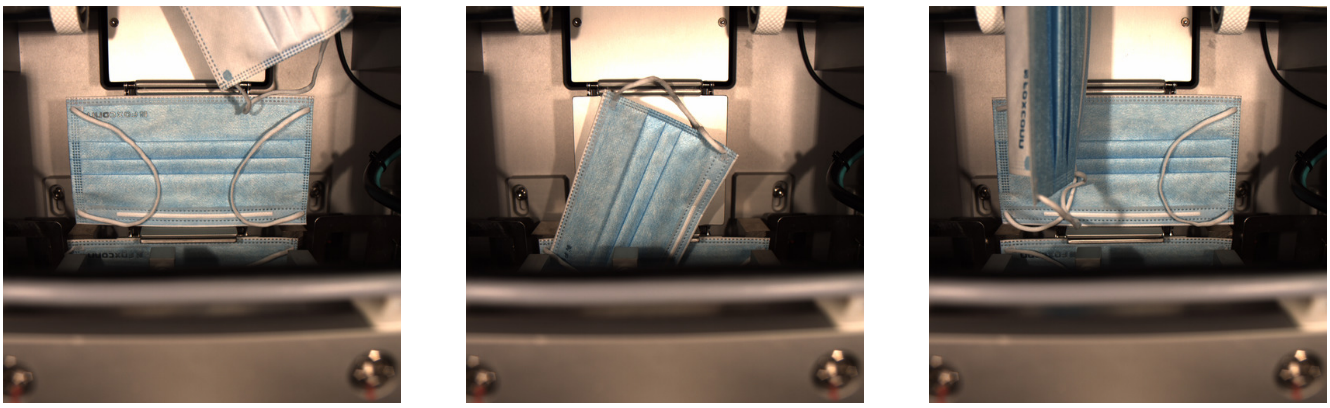

Figure 5: These images where rejected by the classifier

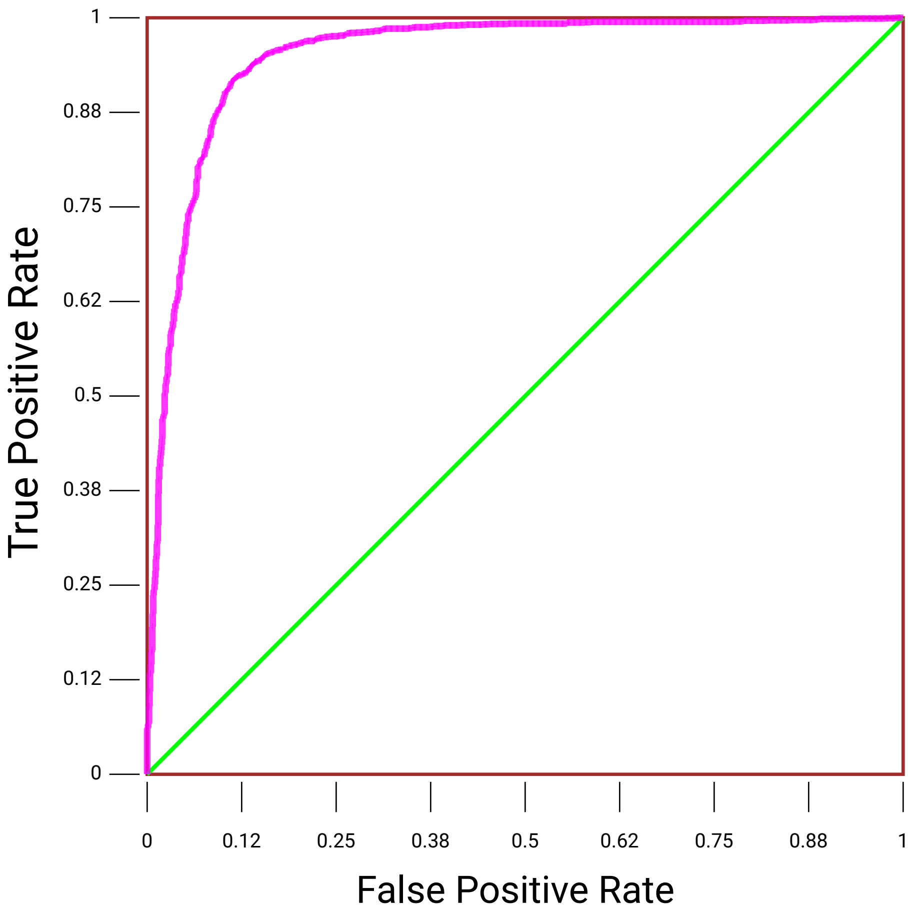

Figure 6: The Receiver Operating Characteristics (ROC) of the Classifier

Data, machine learning can be used to solve problems, without precise definitions, using only human experts to provide supervision during the training process.
We have successfully managed to apply the SVM algorithm with the Haar-like features to provide a QA solution in a surgical mask production line. Our findings support the idea that Machine Learning has much unused potential in industrial applications.

References

[1] Nurul Iman Saiful Bahari, Asmala Ahmad, and Burhanuddin Mohd Aboobaider. Application of support vector machine for classification of multispectral data. *IOP Conference Series: Earth and Environmental Science*, 20:012038, jun 2014.
[2] G. Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools*, 2000(120):122–125, 2000.
[3] Nishant Chauhan and Byung-Jae Choi. Performance analysis of classification techniques of human brain mri images. *INTERNATIONAL JOURNAL of FUZZY LOGIC and INTELLIGENT SYSTEMS*, 19:315–322, 12 2019.
[4] Anoopa Chittilappilly and Kamalraj Subramaniam. Svm based defect detection for industrial applications. In *2017 International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 1–5, 01 2017.
[5] Qiong Dai, Jun-Hu Cheng, Da-Wen Sun, and Xin-An Zeng. Advances in feature selection methods for hyperspectral image processing in food industry applications: A review. *Critical reviews in food science and nutrition*, 55, 04 2014.
[6] Dmitriy Fradkin and Ilya Muchnik. Support vector machines for classification. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 70(1), 01 2006.
[7] Daniel Garten, Katharina Anding, and Steffen Lerm. Image based recognition - recent challenges and solutions illustrated on applications. *Machine Learning and Applications: An International Journal (MLAIJ)*, 3(3), 2016.
[8] Souhail Guennouni, Ali Ahaitouf, and Anass Mansouri. A comparative study of multiple object detection using haar-like feature selection and local binary patterns in several platforms. *Model. Simul. Eng.*, 2015, January 2015.
[9] Hans J. Johnson, M. McCormick, L. Ibáñez, and The Insight Software Consortium. *The ITK Software Guide*. Kitware, Inc., third edition, 2013. *In press*.
[10] Menyhárt József and Róbert Szabolcsi. Support vector machine and fuzzy logic. *Acta Polytechnica Hungarica*, 13(5):205–220, 01 2016.
[11] Ziqiu Kang, Cagatay Catal, and Bedir Tekinerdogan. Machine learning applications in production lines: A systematic literature review. *Computers and Industrial Engineering*, 149:106773, 08 2020.
[12] Atibi Mohamed, Atouf Issam, Boussaa Mohamed, and Bennis Abdellatif. Real-time detection of vehicles using the haar-like features and artificial neuron networks. *Procedia Computer Science*, 73:24–31, 2015.
[13] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1226–1238, August 2005.
[14] Johnny Saldaña. *The Coding Manual for Qualitative Researchers*. SAGE Publications Ltd., 1 Oliver's Yard 55 City Road, London, UK, 3rd edition, 2016.
[15] E. Stewart. *Intel Integrated Performance Primitives: How to Optimize Software Applications Using Intel IPP*. Intel Press, 2004.
[16] Alberto Tellaeche, Gonzalo Pajares, Xavier P. Burgos-Artizzu, and Angela Ribeiro. A computer vision approach for weeds identification through support vector machines. *Applied Soft Computing*, 11(1):908–915, 2011.
[17] Paul Viola and Michael J. Jones. Robust real-time face detection. *Int. J. Comput. Vision*, 57(2):137–154, May 2004.
[18] Yu Wu and Yanjie Lu. An intelligent machine vision system for detecting surface defects on packing boxes based on support vector machine. *Measurement and Control*, 52(7-8):1102–1110, 2019.
[19] Shaojie Yang, Xiang Li, Xiaodong Jia, Yinglu Wang, Haodong Zhao, and Jay Lee. Deep learning-based intelligent defect detection of cutting wheels with industrial images in manufacturing. *Procedia Manufacturing*, 48:902–907, 2020. 48th SME North American Manufacturing Research Conference, NAMRC 48.
[20] Jiandong Zhao, Hongqiang Wu, and Liangliang Chen. Road surface state recognition based on svm optimization and image segmentation processing. *Journal of Advanced Transportation*, 2017, 07 2017.