Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 196 (2022) 217–225

Procedia Computer Science

www.elsevier.com/locate/procedia

CENTERIS - International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies 2021

# A Ubiquitous Service-Oriented Automatic Optical Inspection Platform for Textile Industry

Dibet Garcia Gonzalez[a,*] Yusbel Chavez Castilla[a], Somayeh Shaharadaby[a], Ana Mackay[a], Lúcia Soares[a], Pedro Guimarães[a], Francisco Morais[a], Joel Puga[a], Filipe Meneses[a,b], Adriano Moreira[a,b], Miguel Machado[c], Telmo Adão[a], Luís Magalhães[a], Miguel Angel Guevara López[a]

[a]Center for Computer Graphics, Campus de Azurém, Edifício 14, Guimarães, 4800-058, Braga, Portugal
[b]Algoritmi Research Center, University of Minho, Guimarães, 4800-058, Portugal
[c]ERT Group. Av 1o Maio – Z.I. Travessas. Apartado 181. 3701-911, São João da Madeira, Portugal

**Abstract**

Within a highly competitive market context, quality standards are vital for the textile industry, in which related procedures to assess respective manufacture still mainly rely on human-based visual inspection. Thereby, factors such as ergonomics, analytical subjectivity, tiredness and error susceptibility affect the employee's performance and comfort in particular and impact the economic healthiness of each company operating in this industry, generally. In this paper, a defect detection-oriented platform for quality control in the textile industry is proposed to tackle these issues and respective impacts, combining computer vision, deep learning, geolocation and communication technologies. The system under development can integrate and improve the production ecosystem of a textile company through a properly adapted information technology setup and associated functionalities such as automatic defect detection and classification, real-time monitoring of operators, among others.


Peer-review under responsibility of the scientific committee of the CENTERIS –International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies 2021

*Keywords:* Automatic Optical Inspection; Computer Vision; Deep Learning; Textile Industry; Indoor Positioning; Microservices Architecture.

* Corresponding author. Tel.: +351-932-170-505
*E-mail address:* dibet.gonzalez@ccg.pt


Peer-review under responsibility of the scientific committee of the CENTERIS –International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies 2021
10.1016/j.procs.2021.12.008

## 1. Introduction

In today's competitive market, industries struggle to minimise defects in their final products. For this purpose, identifying the defect and finding a solution to prevent its recurrence is essential. The traditional methods for defects inspection rely on human eye-based analysis, which is time-consuming and prone to errors, and it negatively impacts employees' comfort and ergonomics. Thereby, in recent years, Automatic Optical Inspection (AOI) and its remarkable capabilities for detecting manufacturing defects in a wide range of industrial activities have been arousing the attention of the scientific community and industry-related professionals.

A large spectrum of approaches supports AOI systems/platforms such as traditional computer vision methods, classic classification approaches such as Support Vector Machine (SVM), Artificial Neural Networks (ANN), etc. Recently, Deep Learning (DL) algorithms and methods have been modernizing computer vision in several tasks (e.g., classification, detection, segmentation) and are becoming increasingly popular in applications regarding quality assurance AOI.

In the context that characterizes the textile industry, quality standards in activities that include – but are not restricted to – production is getting increasingly tight and meticulous in meeting the demanding requirements that have been laid down by the satellite business models (automotive industry, fashion/clothing commerce, etc.). Considering this issue allied to the available technological resources that unlock pathways to AOI, this paper proposes a defect detection-oriented platform for quality control in the textile industry, combining computer vision, deep learning, geolocation and communication technologies. Furthermore, the developed system can integrate and improve the IT ecosystem of a company through increased functionalities such as automatic defect detection and classification, real-time monitoring of operators, among others.

This work is organised as follows: Section 2 includes background about service-oriented architectures, textile defect inspection algorithms and methods, and indoors location; the proposed textile inspection platform, which is divided into modules, is presented in Section 3. Then, section 4 shows some results, and finally, some conclusions are drawn in section 5.

## 2. Background on service-oriented architectures, indoors location and textile defect inspection.

Background on relevant technologies for service-oriented AOI focusing infrastructure perspective, vision algorithms for defect inspection and indoors tracking/mobility will be provided next.

### *2.1. Service-Oriented Architectures*

Regarding the reference architecture model for industry 4.0, flexible communication and a distributed system architecture are key features of future production systems [1]. All instances that participate in the value creation must be connected and distribute information, which can be achieved through service-oriented architecture, defined as a loose coupling of independently managed information systems [2]. Different components implement atomic functionalities in this followed approach, and the composition of all these components forms a complete architecture ecosystem.

Each microservice implements a specific end-to-end business domain or capability within a given context boundary, and each must be developed autonomously and deployed independently. Also, we followed the database per service pattern, ensuring that the services are loosely coupled, and changes to one service's database do not impact any other services. This way, each service uses the type of database that best suits its needs, allowing us to implement SQL and NoSQL data models.

Such an architecture results in a many-to-many communication network, where two typical models exist to enable communication between participants: the client/server and the peer-to-peer models. This way, it supports Event Driven SOA features so that different components can decide their interaction pattern and react to internal and external events. And components can behave either as services or event producers and consumers.

The services must handle requests from the application's clients and collaborate to handle those requests. They also need to synchronously communicate results in tight runtime coupling, requiring both the client and the service to be available for the duration of the request. So, they need a Remote Procedure Invocation (RPI) technology for inter-service communication.

Industrial IoT requires flexible communication and a distributed, loosely coupled system architecture. Using the peer-to-peer communication model combined with message-oriented middleware (MOM) is a common approach for these new systems [2], which was also followed in this project.

### *2.2. Defect Detection Methods*

A defect detection process based on computer vision techniques follows a workflow that typically includes a pre-processing step and ends with a classification step.

One pre-processing method uses a filter to reduce the noises, e.g., Bilateral, Wiener, Gaussian, Median, Mean, Laplacian. In [3, 4], the authors try to determine which filter better suits eliminating noises. After the noise removal, the most straightforward approach to getting useful information from the image is to use thresholding, a non-contextual approach.

This method is based on a threshold value to turn a grey-scale image into a binary image. Mehmet Sezginet al. [5] have compared 40 selected thresholding methods from various categories in non-destructive images based on the combined performance measures.

Other approaches for defect inspection are based on the image classification process. S. Sadaghiyanfam [6], used Gray Level Co-occurance Matrix (GLCM) and wavelet transform for detect inspection and compared the results. They used three types of wavelet, including Haar, 2nd order Daubechies (db2), Sym4 with the different number of levels. The combination of GLCM with the other methods shows a significant result in defect detection. For instance, in [7], they convert the acquired image to a binary image using a Discrete Curvelet Transform (DCT) and then proceeded it by GLCM. P. Deshappriya et al. [8] used a Convolutional Neural network (CNN) to classify the fabric defect and locate it in the input image. Their results showed that using a light beam of similar colour intensity to the original material detected more accurate defect identification than white light. Whereas traditional convolutional network only has connections between each layer, Gao Huang et al. [9] proposed Dense Convolutional Network (DenseNet), which connects each layer to every other layer in a feedforward fashion. This network (DenseNet) was also used by [10], and they modified it regarding their requirements.

### *2.3. Indoor Location and Operators Tracking*

GNSS – Global Navigation Satellite System is the most well-known positioning solution, working almost anywhere, providing the location in an absolute worldwide referential. However, this technology works only outdoor since it needs a direct line-of-sight between the emitters (satellites) and the GNSS receivers.

Different technologies were explored in the last years to build indoor positioning systems. Among the most used are the ones based on Wi-Fi and BLE networks and UWB - Ultra-WideBand radio. In particular, the UWB makes it possible to implement indoor location with very high accuracy. Although high-cost specific hardware installed every few square meters are required, becoming too expensive and complex to deploy in a small or medium warehouse or factory.

Wi-Fi networks are very popular, low cost and globally available. The radio signal of these networks can be used to build positioning systems in an indoor environment. Several techniques and algorithms have been developed that generally fall into three broad categories: proximity, triangulation and scene analysis [11]. Scene analysis techniques are based on the principle that radio measurements are location dependent.

These techniques are composed of two phases. In the first phase, a radio map is constructed, which contains fingerprints collected from landmarks. In the second phase, called online, a user device collects a fingerprint which, by comparison/similarity with those existing on the radio map, allowing the position to be estimated.

As an alternative, or in addition to Wi-Fi, indoor positioning solutions can be built based on BLE. BLE transmitters emit a radio signal with a short-range (compared, for example, to Wi-Fi), making it possible to create positioning solutions with potentially greater accuracy. Most of the time, BLE emitters have to be explicitly installed for implementing indoor positioning systems. However, BLE emitters are low cost; they can be installed easily because they are small in size, can easily be glued to objects, and when powered by batteries, they work for months. Based on current state-of-the-art, Wi-Fi and BLE are the most convenient technologies to deploy scalable indoor positioning systems in ample spaces, demanding a reduced or almost null investment.

## 3. The ubiquitous AOI system for textile defects detection.

We used a Service-Oriented Architecture (SOA) to solve the context of the business domain. Microservices are a modern interpretation of the service-oriented architecture and were used to build a server application as a set of small services.

Use Case models allow to represent the visualization of a system's process and can be considered as a method of gathering requirements, namely in the creation of a context for the gathering of requirements and subsequent derivation of logical architecture diagrams of the process. This way, user and system requirements were separated into functional and non-functional requirements. And the mapping of processes using the National Institute of Standards and Technology Cloud Computing Reference Architecture (NIST CCRA) [12] reference model allowed us to verify the suitability of cloud computing to the proposed solution. We used a context model to describe the scope of work, its adjacent systems, and the data that flows between them. The interfaces between adjacent systems and the context of the work to be carried out indicate why they are identified in the Client industrial context.

This way, the delivery model in this project is a software as a service, in a cloud infrastructure operated as a private cloud (AOI Platform). In the private cloud, the infrastructure and computing resources are available from Client in-house IT team, which operates, installs and manages the computing resources. And the SaaS application in the private cloud is implemented directly on the cloud resources (virtual machines) of the client internal IT team.
The AOI platform provides devices, information and users management features of the product defect detection system on a factory floor. These features include algorithms and methods developed to improve the quality control process in the textile industry. The platform implemented in a microservices architecture is grouped into three modules (Fig. 1): Communication, Inspection and Operator Tracking. Moreover, the platform can interact with external components such as an Enterprise Resource Planning system.

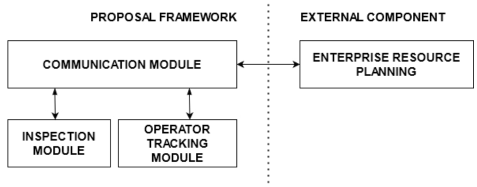


Fig. 1. Global architecture of the proposed platform.

### *3.1. Communication Module - CM*

Features such as authentication, authorization, data storing, retrieval and transmission for other system modules, as well as protocol conversion for interoperability with internal/external systems are implemented in the communication module (Fig. 2). All components implement and publish a REST interface allowing the synchronously exchange of data with the system messaging bus. Moreover, clients of the microservices-based application access the individual services using API gateway pattern in its variation Backends for frontends [13], providing a single-entry point for all clients. The API gateway handles requests simply proxied/routed to the appropriate service. Also, exposes a different API for each client, providing the ones that are best suited to its requirements. As a result of this adoption, we got a simple request/reply system with no intermediate broker. In such a distributed system deployment, services run at fixed, well-known locations (hosts) and can easily call one another using HTTP/REST.

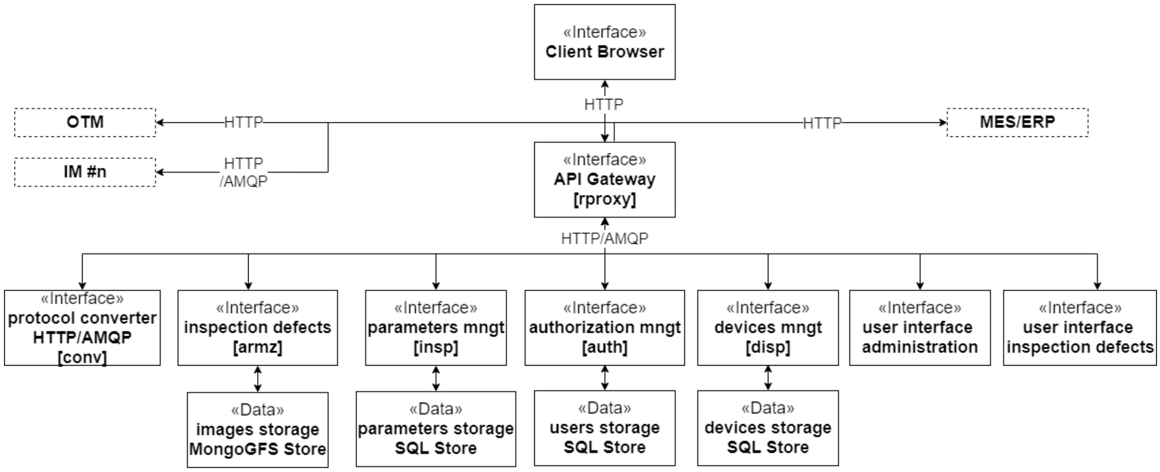


Fig. 2. Components of the Communication Module.

Integration with legacy systems often lacks technological compatibility, which leads to the need for customising integrations to favour interoperability. It is the case of the integration between the IM and an external Enterprise Resource Planning (ERP) system in this project. To overcome this issue, a converter service was modelled to handle the conversion of AMQP protocol between the MOM of the IM, and the HTTP interface of the ERP system. Such a service enables bridging the asynchronous exchange of data with the MOM, decoupling the communication and allowing the parallelisation with several cyber-physical IM. And at the same time, bridging the synchronous exchange of data with ERP, in tight runtime coupling, which requires both the service and the ERP to be available for the duration of the request, and so, be limited by a time-out to handle MOM asynchronous availability.

### *3.2. Inspection Module - IM*

This module includes five components: 1) Image Acquisition (IAC), 2) Defect Detection (DDC), 3) Defect Classification (DCC), 4) Message Broker (MBC), and 5) Visualisation Interface (VIC). Each component represents a specific step in the defect detection workflow. The interaction between these components is shown in Fig. 3.

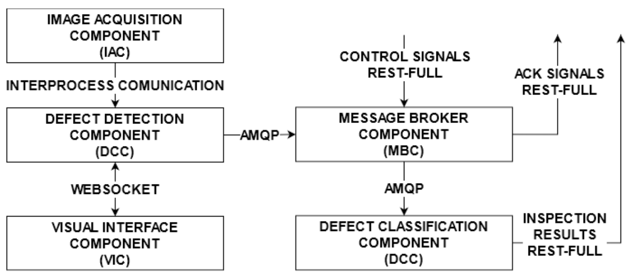


Fig. 3. Components of the Inspection Module and its interaction.

The IAC design allows detecting defects with a minimum size of 1mm, considering a translation speed of 25m/s. In this sense, the selection of linear cameras, lenses and illumination allow representing 1mm with 8 pixels and covers a field of view of up to 1800mm. This setup enables the visualisation of small defects. Still, it increases the amount of data to be processed, e.g. a material with dimensions of 500mm x 1800mm occupies 192MB in RGB colour space. Fig. 4 shows a scheme with the main components of the IAC. LC1 and LC2 are two colour lineal cameras with 7,04 µm as pixel size and 8192 pixels by line (Teledyne Dalsa Camara Linea Color 7.04um 8192x2-48kHz-Color-CMOS-mCL). They are connected to a processing unit (Industrial PC Nuvo 8208GC i7-9700TE with Nvidia RTX 3080) through 2 frame grabbers (Xtium-CL MX4). The illumination equipment includes a white led bar with an adjustable tilt angle.

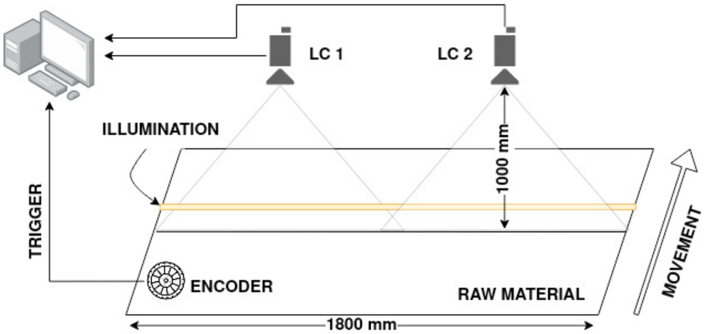


Fig. 4. Scheme of the equipment used in the IAC.

The DDC can handle this amount of data and detect defects based on a set of pre-selected features and a pre-trained artificial neural network (ANN) algorithm. The features are specific to each type of material, and its computation is optimised to run over the CUDA architecture and ensure real-time processing. The DDC is supported by intensities and textures features. The training process of the artificial neural network is executed in advance (offline) from previously annotated databases created from the material of interest.

After the defect detection step, the DDC pushes its output (i.e., defect position and current image) into a queue for classification purposes. Next, the DCC pops the image and performs defect type recognition tasks, resorting to deep learning algorithms. The push and pop processes occur in the MBC, responsible for managing the information flowing to and from the CM. The VIC (Fig. 5 a and b) visualise the acquired images and defects detected in real-time, allowing to supervise the inspection process.

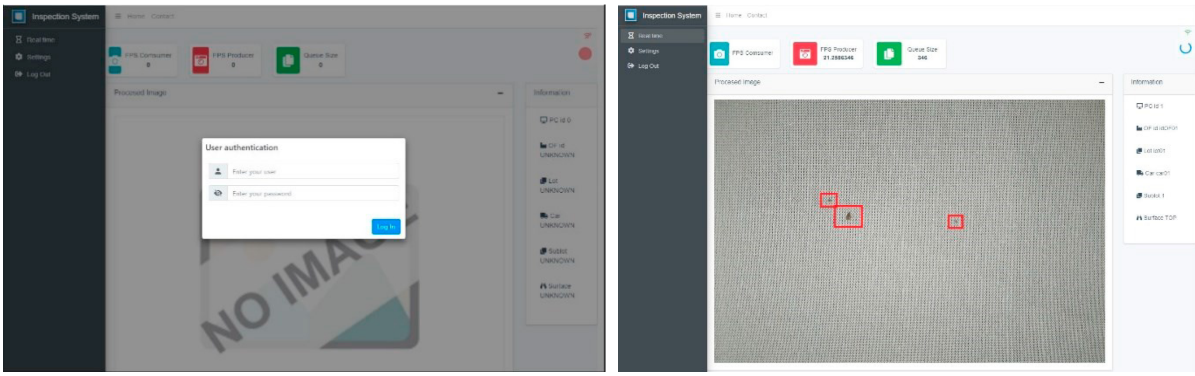


Fig. 5. Visual Interface Component. Left side: login interface, right side: real-time defect detection.

Regarding the DL approach adopted to perform defect detection, SSD Inception V2 is considered, since it guarantees an acceptable trade-off between speed and accuracy [14]. Two main structures were combined to build this DL proposal: SSD [15] and Inception [16]. While the former method is designed for detecting objects based on multibox fitting processes, the latter consists in a supporting mid-layer that potentiates robust feature extraction.

### *3.3. Operator Tracking Module - OTM*

This module is formed by three different components, a mobile app, an indoor location web service, and an intermediary web service in charge of logging, post-processing and routing between the different parts of this and other modules. Communication between components of this module is achieved through a MQTT broker (Mosquitto). All components run on docker containers.

When the inspection module finds a possible but unknown defect, it sends a request to this module's intermediary web service. This communication is made through a specific topic on the MQTT broker. When the intermediary web service receives the request, this is stored in a MongoDB database for logging and history keeping purposes. It then sends a request to the location web service to obtain a list of operators on the factory floor. The request is then routed to the mobile app running on all the returned operators' devices. Afterwards, the operator's response is sent back to the intermediary web service, which logs it and then routes it to a communications module API endpoint for further treatment.

The mobile app, developed with React Native, allows an operator to login into the system (a valid account, created through the global system user management module, is required). It then starts listening, in the background, to a broker topic reserved for the logged-in username.

Considering an industrial context, where operators can be called upon to intervene at any time and therefore may be on the move, a mobile app is the most suitable solution, as a smartphone can be more easily carried. Furthermore, with a focus on usability, a simple app design interface supports the system and facilitates inspection while minimising cognitive load by making available to the user only the elements of interaction necessary to reject or categorise a defect.

When an unknown defect notification arrives, the app alerts the operator (haptic and sound notifications are used) and allows him, through two different screens, to analyse the unknown defect picture, dismiss it as not a defect or categorise it if it is, indeed, a defect. Most of the screen is used to display the image of the possible defect, to facilitate visual inspection and reducing the probability of errors occurring in the operator's analysis. The strategic use of green coupled with the label "yes" and the tick symbol contrasts with the chosen red colour, associated with a "no" label and an X-like cross whenever a confirmation or rejection is needed. The categorisation is performed through a drop-down menu for an easier and more efficient task, and feedback messages are delivered after each completed action. Moreover, the available screens show an informative note that reminds the operator that he/she must go to the machine for a new visual inspection and subsequent decision-making, in case of uncertainty.

## 4. The inspection platform results

Concerning the automatic optical inspection system, it is noteworthy to highlight that the functional requirements regarding the speed of movement of the material to be inspected were met.

The system loading experiments confirmed that the images formed from the two linear cameras could be produced at 4fps, and with a size of 16384x1024 pixels (8pixels by mm). This ensures that the translation speed is higher than 25m/s, set as the maximum translation speed value. The Defect Detection Component can process this information flow and ensure that all images are processed in real-time and without delays.

In addition, the Message Broker component manages to queue all of the detected defects to be classified based on deep learning algorithms. This processing can be carried out in parallel to the processing or afterwards to issue reports about the production quality. Fig. 6 shows some colour defects as examples of the achieved results from deep learning-based algorithms. The inference model used in the defect classification component was trained for 2 classes: colour defect and mechanical defect.

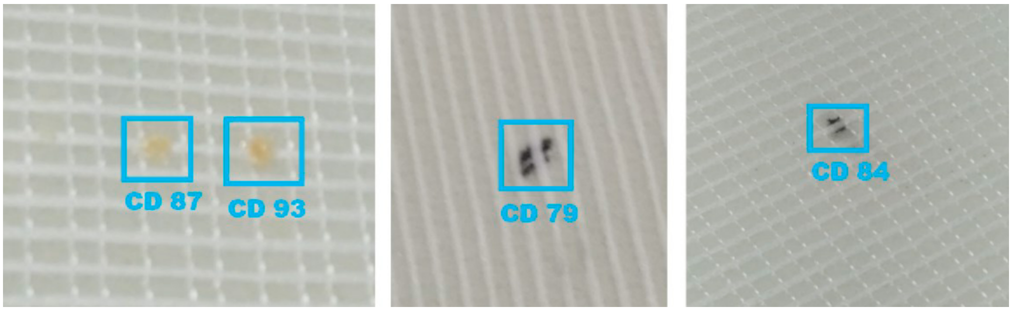


Fig. 6. Defect classification results based on deep learning algorithms.

However, there may be defects that cannot be identified. In this sense, the geolocation module and a tool (mobile application) will allow a remote user to be notified of an unknown defect type. As described in subsection 3.3, this notification is presented on an Android smartphone (Fig. 7) and includes a picture of the defect and the identification of the machine that performed the detection. The remote user uses this information to classify the defect directly from his/her smartphone as a part of an active learning strategy.

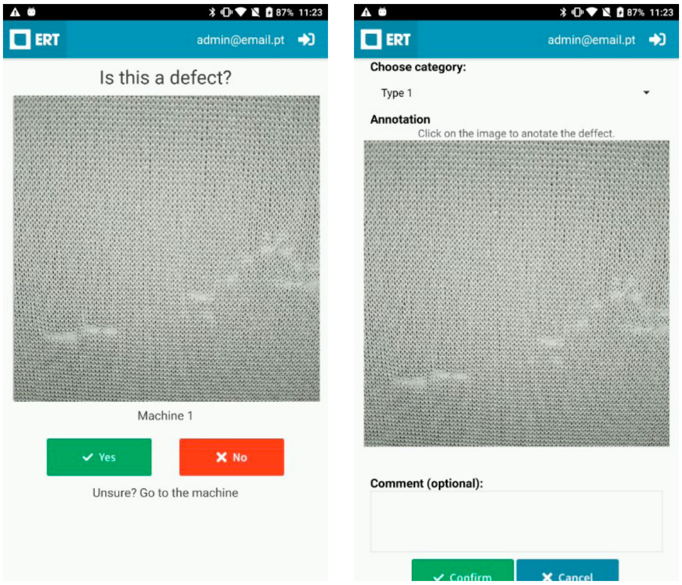


Fig. 7. Unknown defect classification viewed in Android app.

Considering the technology taker perspective, the following main advanteges of using the proposed platform can be highlighted: i) the system configured to inspect one surface or both surfaces ii) detection of defects larger than 1mm square can be performed, due to the high resolution of the integrated cameras, iii) detailed reports can be generated based on the position and classification of each defect, and iv) inter and intra human observer subjectivity is avoided. To ensure that both surfaces are inspected, the system must be duplicated by using two processing units with 4 frame grabbers and 4 linear cameras. Moreover, acquisition and illumination equipments mirrored in relation to the area of interest allow to cover the back part of the coating that also requires inspection, but is out of the human operator sight.

## 5. Conclusions

In this work, a defect detection-oriented platform for quality control in the textile industry was presented. The proposed solution combines computer vision, deep learning, geolocation and communication technologies, enhancing the quality inspection in any enterprise IT ecosystem. As a novelty, the platform: (1) introduces a processing strategy to manage a vast amount of data in real-time and detect small defects, and (2) represents a solution supported in Industry 4.0 that enhance the quality control process. Overall, the current platform succeeds in the objective of detecting most of the defects, even the small ones, while the raw material is moving at high speed (25m/s) processing in real-time.

In a near future, an active learning strategy will be implemented for the continuous enhancement of the current defect-aware deep learning models, under a cyclic process that relies in factory floor skilled labour confirmations upon defects inferred (by machine) bellow a predefined certainty threshold.

## Acknowledgements

This work was financed by the project “Smart Production Process” (No. POCI-01-0247-FEDER-045366), supported under the Incentive System for Research and Technological Development - Business R&DT (Individual Projects).

## References

[1] M. F. P. Sommer, F. Schellroth, J. Schlechtendahl, Message-oriented middleware for industrial production systems, 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), pp. 1217-1223 (2018).
[2] S. Mankovski, Service oriented architecture, in Springer reference, Encyclopedia of Database Systems, L. LIU and M. T. Ozsu, Eds., New York: Springer, 2009, pp. 2633–2636 (2019).
[3] Z. Ramadan, Optimum image filters for various types of noise, Telkom- nika (Telecommunication Computing Electronics and Control) 16 (2018) 2458–2464. doi:10.12928/TELKOMNIKA.v16i5.10508.
[4] M. Gupta, H. Taneja, L. Chand, Performance enhancement and analysis of filters in ultrasound image denoising, Procedia Computer Science 132 (2018) 643 – 652, international Conference on Computational Intelligence and Data Science. doi:https://doi.org/10.1016/j.procs.2018.05.063.
[5] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging 13 (1) (2004) 146 –165. doi:https://doi.org/10.1117/1.1631315.
[6] S. Sadaghiyanfam, Using gray-level-co-occurrence matrix and wavelet transform for textural fabric defect detection: A comparison study, in: 2018 Electric Electronics, Computer Science, Biomedical Engineerings’ Meeting (EBBT), 2018, pp. 1–5. doi:10.1109/EBBT.2018.8391440.
[7] A. P, R. S. Sabeenian, Fabric defect detection using discrete curvelet transform, Procedia Computer Science 133 (2018) 1056 – 1065, international Conference on Robotics and Smart Manufacturing (RoSMa2018). doi:https://doi.org/10.1016/j.procs.2018.07.058.
[8] P. Bandara, T. Bandara, T. Ranatunga, V. Vimarshana, S. Sooriyaarachchi, C. D. Silva, Automated fabric defect detection, in: 2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer), 2018, pp. 119–125. doi:10.1109/ICTER.2018.8615491.
[9] G. Huang, Z. Liu, L. V. D. Maaten, K. Q. Weinberger, Densely con- nected convolutional networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2017, pp. 2261–2269. doi:10.1109/CVPR.2017.243.
[10] Z. Zongwei, G. Han, G. Jia, L. Shu, Modified densenet for au- tomatic fabric defect detection with edge computing for minimizing latency, IEEE Internet of Things Journal PP (2020) 1–1. doi:10.1109/JIOT.2020.2983050.
[11] H. Liu, H. Darabi, P. Banerjee, J. Liu, Survey of wireless indoor posi- tioning techniques and systems, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37 (6) (2007) 1067– 1080. doi:10.1109/TSMCC.2007.905750.
[12] Liu, F., Tong, J.; Mao, J.; Bohn, R.; Messina, J.; Badger, M.; Leaf, D. Nist Cloud Computing Reference Architecture, 2011.
[13] Chris Richardson, Applying the microservice architecture pattern lan- guage, https://microservices.io/articles/applying.html, Last accessed on 2021-06-02 (2021).
[14] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I Fischer, Z. Wojna, Y. Song, S. Guadarrama, K Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7310--7311, 2017.
[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector.InEuropean Conference on Computer Vision, pages 21–37.Springer, 2016.
[16] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke and Andrew Rabinovich. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.