

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Distributed Analytics Framework for Integrating Brownfield Systems to Establish Intelligent Manufacturing Architecture

Vigneashwara Pandiyan and Wahyu Caesarendra

Abstract

Intelligent manufacturing otherwise called as smart manufacturing concentrates upon optimising production and processes by making full use of data available. It is regarded as a new manufacturing model where the entire product life cycle can be simplified using various smart sensors, data-driven decision-making models, visualisation, intelligent devices, and data analytics. In the Industry 4.0 era, Industrial Internet of Things (IIoT) architecture platform is required to streamline and secure data transfer between machines, factories, etc. When certain manufacturing industry is equipped with this platform, an intelligent manufacturing model can be achieved. In today's factories, most machines are brownfield systems and are not connected to any IoT platforms. Thus they cannot provide data or visibility into their performance, health, and optimal maintenance schedules, which would have improved their operational value. This paper attempts to bridge this gap by demonstrating how brownfield equipment can be IIoT enabled and how data analytics can be performed at the edge as well as cloud using two simple use cases involving industrial robot on the abrasive finishing process. The focus of this paper is on how a scalable data analytics architecture can be built for brownfield machines at the edge as well as the cloud.

Keywords: deep learning, digital mechanical system, industry 4.0, machine learning, smart manufacturing

1. Introduction

The Industrial Internet of Things (IIoT) is set to change the face of manufacturing. Many industries are already gaining from digitalisation journey and Internet of Things (IoT), and industrial manufacturing is leading the way [1–3]. The IIoT is considered to be a modern manufacturing concept under Industry 4.0 and has been evolving rapidly based on the industrial requirement. A typical IIoT platform should consist of cutting-edge information technology (IT) infrastructure for data acquisition and sharing [4]. The features of an intelligent manufacturing includes real-time data collection and sharing among various manufacturing resources such as machines, subsystems, operators, and materials [5].

The ability to “sensorize” all and extract data to offer insights and forecasts is a huge benefit for manufacturing [6]. The Internet of Things and data remain

inherently linked together. If a system goes down, for example, connected sensors can automatically localise the issue. This process is time-intensive when humans do it manually. Apart from sensorization, ubiquitous connectivity is the crucial pillar of IoT that assures to deliver the value by connecting numerous devices/assets that generate useful data. Acquiring accurate and reliable data from machines and their components is the first step in developing an IIoT architecture. Sensors might directly measure the data, which can also be obtained from controllers. Data might also be acquired from enterprise manufacturing applications such as enterprise resource planner (ERP), manufacturing execution system (MES), supply chain management system (SCMS), etc. [7]. As far as data, two important factors have to be considered. Firstly, selecting proper sensors (type and specification) for the specific objective is critical. Secondly, a seamless and tether-free method to manage the acquisition and transfer data considering various data types to the central server is required.

The data generated from IoT devices turn out to be of value only if it gets subjected to analysis, which brings data-driven analytics into the architecture. Data analytics (DA) is defined as a process, which is used to examine big and small data sets with different data properties to extract meaningful conclusions and actionable insights. These conclusions of data analysis are usually in the form of trends, patterns, and statistics that aid in effective decision-making processes. Data analysis requires support such as hardware resources such as GPU and servers for computing. The frequency of using the results of analytics may vary from real-time problems such as anomaly detection and tool wear [8, 9] to long intervals, such as predicting remaining useful life [10]. Depending on the requirement, analytics can be performed locally, i.e. edge or remotely, i.e. the cloud.

According to the aforementioned current problem statement, the IIoT-based cyber-physical system for manufacturing can fill this gap. One typical architecture of the IIoT-based cyber-physical system for manufacturing is shown in **Figure 1**. The

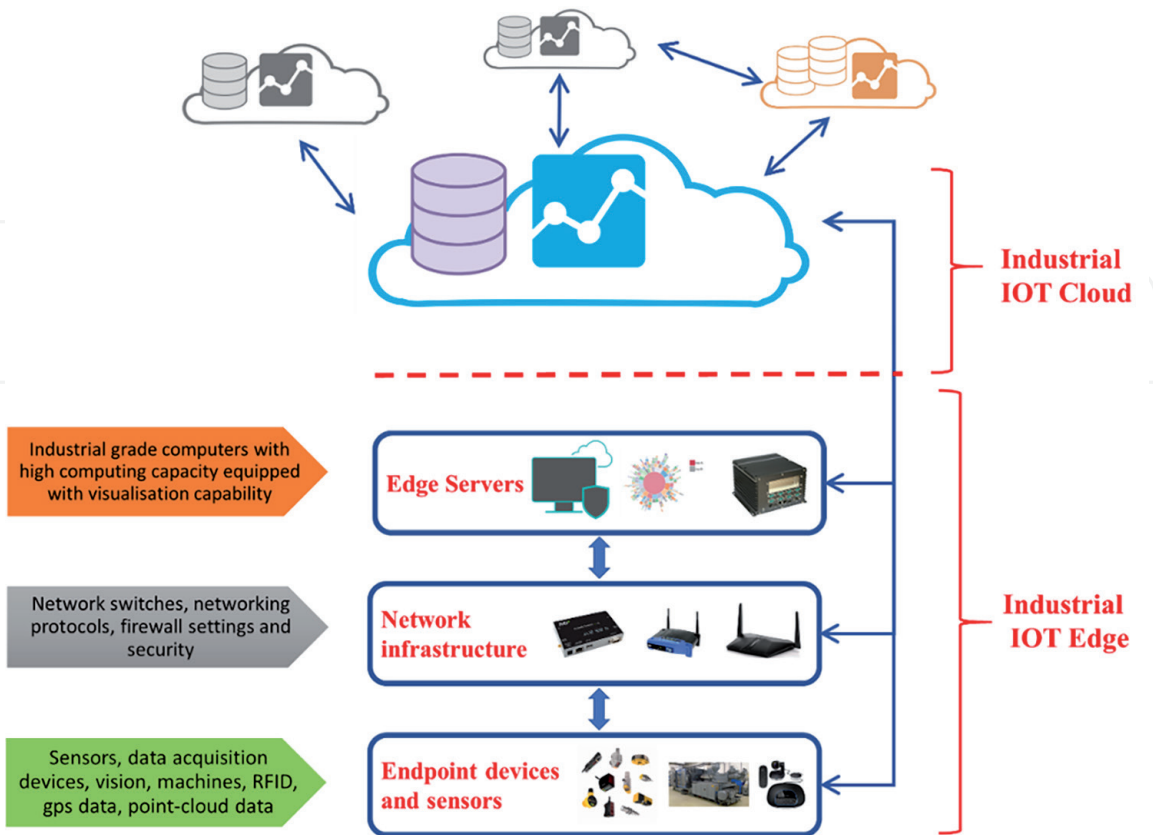


Figure 1.
Architecture of IIoT-based cyber-physical system for manufacturing [11].

first IoT layer captures data from IoT devices such as geographic coordinates, RFIDs, sensor signals, or other unstructured data. With the help of advanced computer networks, such as cyber (or software), resources and tools can be integrated with the manufacturing equipment. Connectivity is what enables the three pillar devices, data, and analytics to work. Devices must be interconnected to transfer data. Data cannot be sent or received without reliable high-bandwidth connectivity that supports real-time data flow from the many devices living on a network. Without connectivity, there would be no data to analyse and no analytics available to optimise systems and create efficiencies. IoT deployments would lack scalability. A standardised communication protocol is always required for data exchange. Many IIoT protocols can be used for networking machines, such as the Message Queuing Telemetry Transport (MQTT) protocol and the Constrained Application Protocol (CoAP). However, connectivity also exposes industrial devices to security attacks which not only disrupt entire systems but can also pose safety risks. An IIoT platform must provide security to minimise risks and keep operations protected from physical breaches and cyberattacks, by monitoring the behaviour of all data sources and alerting operators when anomalies are detected in the manufacturing environment. Securing end-to-end IIoT systems is critical in order to avoid unwanted financial and safety consequences.

The data are then stored and managed in the edge layer by IIoT platform. Like said before, it is the processing of the data acquired and stored that makes the factory realise the importance of digitalization. Data processing ranges from simple visualisation in the dashboard screen to complex cutting-edge data-driven algorithm output. High-end server or industrial PCs are generally used in this layer. Because of the cloud's ability to house large amounts of data, they are a key pillar in IIoT architecture. Today, the potential of using cloud technologies for advanced manufacturing is very high. Cloud computing can be viewed as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources including networks, storage, services, and servers. Cloud computing, as well as IoT, works towards increasing the efficiency of everyday tasks, and both have a complementary relationship [12]. On the one hand, IoT generates lots of data, while on the other hand, cloud computing paves the way for this data to travel [13]. Many cloud providers take advantage of this to provide a pay-as-you-use model where customers pay for the specific resources used.

In today's factories, most of machines/systems are not connected (brownfield) and thus cannot provide data or visibility. Multinetwork environments that include ageing machines of different types and software do not speak on common connectivity language. These two pose significant challenges for many IIoT solutions and hinder the implementation of data analytics. A thorough assessment, planning, scoping, and later execution are required for implementing and performing analytics for such machines to be a part of IIoT architecture, which will be demonstrated in this research work with two use case studies.

The paper is organised as follows. A brief introduction to IIoT architecture is described in Section 1, followed by a brief overview on online quality measurement for manufacturing industries in Section 2. The experimental setup and IIoT architectural design for data analytics are presented in Sections 3 and 4. The results are described in Section 5, followed by conclusions in Section 6.

2. Output quality measurement in manufacturing process

In recent years, the development and optimization of advanced manufacturing processes have been continuously pushed to fulfil the higher demand for performance specification of the components produced. Due to a need for more consistent products

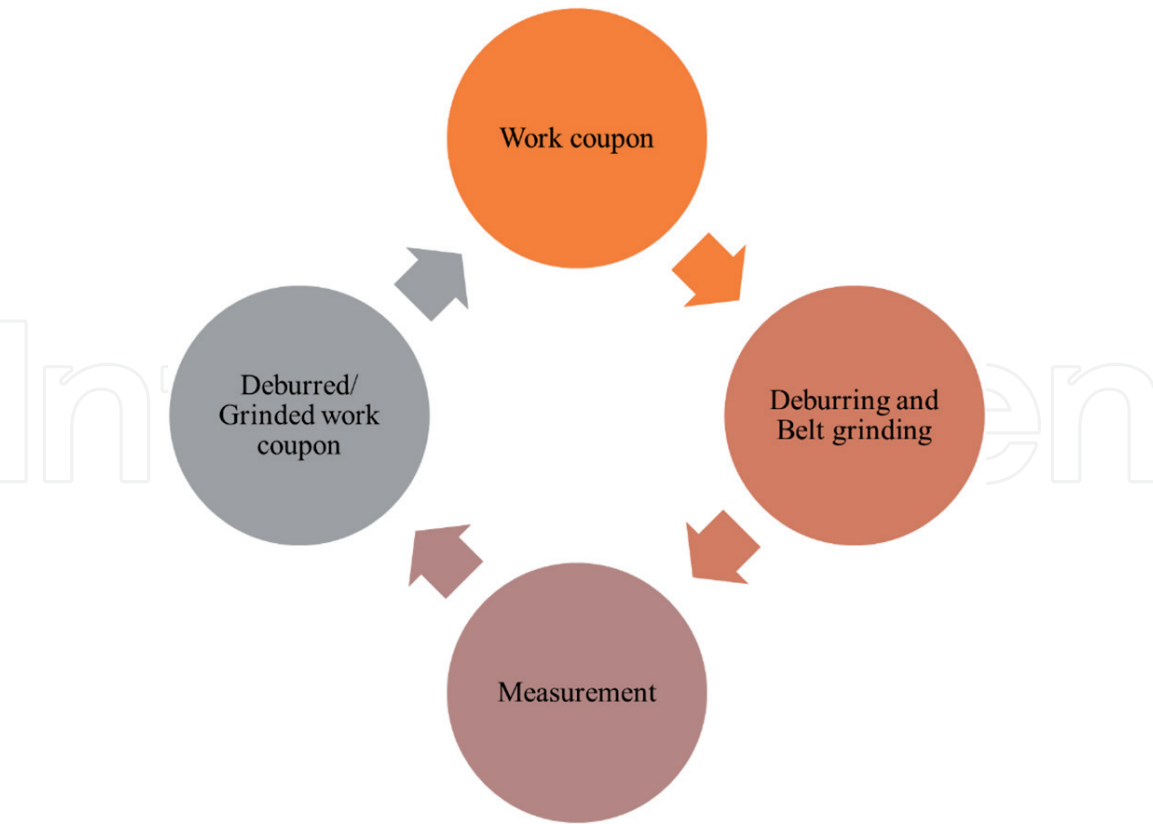


Figure 2.
Deburring/belt grinding cycle in manufacturing.

quality, product variability and product complexity, there is an increase in the manufacturing cycle and product quality measurement time. The reduction of process quality measurement requires decision making by Artificial Intelligence (AI) algorithms for which a data transfer framework for connecting between sensors, robots and devices is required. In other words, the human intervention in manufacturing line will be decreased with the development of connected intelligent manufacturing floors allowing them to concentrate more over the processes optimization, newer product designs and even maintenance. By applying an intelligent architecture to take more control over the manufacturing processes, human inconsistency and measurement time can be reduced and at the same time production capacity can be increased. These architectures will also reduce the scrap materials and enable a cleaner environment. In this paper, two case studies of the evolution of manufacturing processes from the conventional manufacturing line to smart IoT- enabled manufacturing line is presented. The processes used in our case study are deburring and belt grinding.

At present, the manufacturing industries, especially aerospace, are equipped with the industrial robots to perform manufacturing processes such as deburring and belt grinding for surface finishing. However, the output quality monitoring of surface finishing processes such as deburring and belt grinding is accomplished through conventional manual measurement. In high-volume productions, a manual measurement can lead to inconsistency from operator to operator and result in variations in product quality measurement such as thickness and surface roughness. **Figure 2** describes the deburring process cycle which involves conventional output quality measurement. As shown in **Figure 2**, which starts from the work coupon, the cycle only pauses at output quality measurement once the deburring or belt grinding process is executed. During the pause, the robot controller retracts the industrial robot to its home position. After which, an operator starts the measurement process. After the measurement is done, the cycle continues if the surface finish quality measured or required material removal is not achieved.

3. Introduction to cloud computing for smart manufacturing

In cloud computing, there are three categories of services to select from, namely, the Infrastructures a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). The sequence of these three in terms of development foundation can be described in **Figure 3**. PaaS is one layer above IaaS as platform is built on infrastructure and SaaS is one layer above PaaS as software is built on platform. The IaaS provides infrastructure consisting of computer resources and servers with network connectivity and cloud capability. Users have direct access to their servers and storage usually via application programming interface (API). The PaaS is built on a physical server infrastructure and provides a platform where users can build their software on. Meanwhile, SaaS provides ready-to-use software for users that can be accessed anywhere remotely as long as there is Internet connection. The SaaS, for example, are Google Apps, Microsoft Office 365, and Adobe Creative Cloud. Before proceeding further, the category of service has to be selected according to the requirement needed.

In building the Internet of Things (IoT) application, Software as a Service is not a relevant choice to select, unless if there is third-party software found to be useful for the application required. Thus, the selection is between IaaS and PaaS. **Table 1** lists the differences between IaaS and PaaS in the context of manufacturing application.

After considering these differences, a conclusion can be drawn. Over the consideration of long-term commitment, IaaS would be suitable for the scalability to integrate different process-specific applications into one main application. However, the scale of this application is an enterprise-level of application. A research work explains the study of IaaS architecture in more detail which can be used a useful reference for the IaaS level of implementation [14]. Meanwhile, for project level, PaaS is most suitable to build the application on. It is because in PaaS, there is no need to consider infrastructure development. Instead, the focus can be directed towards application development.

In addition to this, what cloud services are available in the market currently? In 2009 a performance comparison of several popular cloud services platform has been done [15]. The selection of cloud services should be adjusted to the time of implementation as cloud services features are always updated and have different advantages for different cloud services provider. The implementation of algorithm conversion to the cloud services provider is much easier to be done now as research platforms such as Python and LabVIEW are currently available. In addition to this difference, it is important to understand the architectural difference between the categories as explained in **Figure 4**. Hence, as PaaS is concerned, the development required to bring deburring into Industry 4.0 is on the applications and data.

The data flow of deburring and belt grinding process that includes the cloud services can be explained in **Figure 5**. In **Figure 5**, Storage Cloud Service (SCS) is

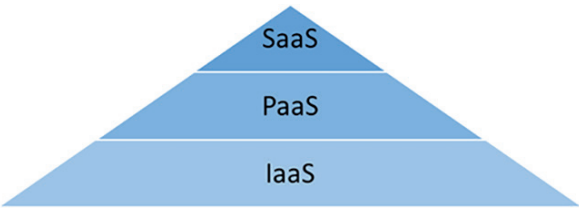


Figure 3.
Cloud services hierarchy.

Criteria	IaaS	PaaS
Developer main focus	Storage, networking, computing	Building application
Platform flexibility to build software	Flexible as developer builds the software on a platform that is on a common infrastructure	Non-flexible as developer builds the software on a platform specific to the provider
Learning curve	Higher learning curve as developer has to build both the platform and the software	Lower learning curve as developer only has to focus on the software
Scalability	Allow more flexibility	Has certain limit of scalability
Intercommunication between software	Allow software of different platforms to communicate easily on the same infrastructures	Software of different platforms must communicate to each other from different infrastructures

Table 1.
Difference between IaaS and PaaS.

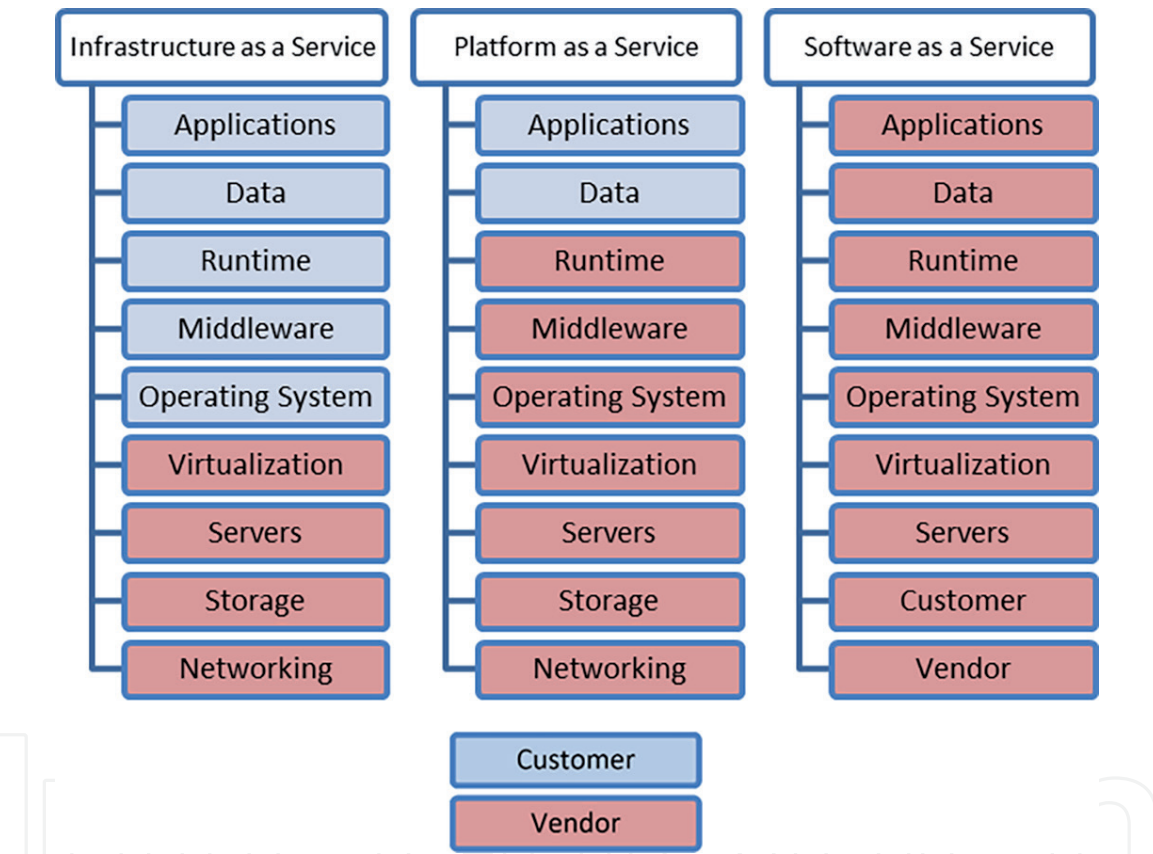


Figure 4.
Architecture difference in cloud services.

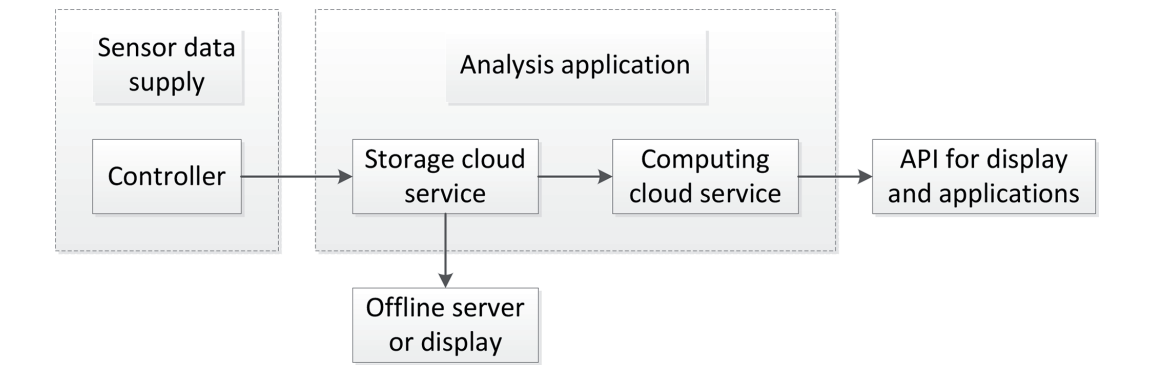


Figure 5.
Deburring and belt grinding data flow conversion from offline to online platform.

Differences	Offline data analysis	Online data analysis
Memory storage and compute capacity	Machine learning performance limited by the built-in storage capacity and memory capacity in the controller	Machine learning performance limited only by the storage and compute quota purchased under the predefined budget
Scalability	Hardware and software scale-up requirement is time-consuming	Only software scale-up requirement is time-consuming
Data resource usage and compatibility	Local data analysis using predefined development environment (e.g. LabVIEW)	Cloud data analysis using more development environment options (e.g. TensorFlow, Kaffee2, Keras)

Table 2.
Implementation difference between offline and online data analysis.

the storage service to store the sensor data and the result in the cloud. Meanwhile, the computing cloud service (CCS) is the computing capacity to process the data. Thus, the machine learning is adapted to the cloud through CCS. This architecture provides API output for display or other applications when required. In addition, the data in SCS is also tapped into offline server or display when required.

The difference in implementation between offline and online data analysis through the same machine learning model is described in **Table 2**, assuming the use of cloud services by third party.

4. Experimental setup (case study)

The robot-assisted manufacturing processes are gaining popularity in industries that move towards automation. Integration of the processes on the robot helps in easing the axis of motion and also ensures that the force applied and repeatability is maintained [16, 17]. Abrasive belt grinding and deburring processes are typically combined with an industrial robot in the manufacturing industry for achieving the desired surface finish and tolerance. In this paper, we will be demonstrating how these two brownfield manufacturing systems can be IoT enabled, and analytics can be performed at two different levels, i.e. cloud and edge. The process data from deburring will be used for edge analytics, and data from belt grinding will be used to perform machine vision analytics at the cloud.

4.1 Deburring

The fundamental of Industry 4.0 is the data communication. In deburring, the main data communication path happens between (1) the sensors to the DAQ (National instrument – Compact RIO) controller, (2) the DAQ controller to the robot controller, (3) robot controller to the ABB robot, and (4) DAQ controller to the server and cloud. The data communication is done through digital and analog input/output with the maximum analog transfer rate at 4 MS/s per module four 16-bit (64 kb) analog input. This translates to maximum of 256 Mb/s data transfer rate per module if the module is fully utilised to its capacity.

Figure 6 explains the data flow in the deburring process. The data transfer occurs locally from the sensors and DAQ controller initially. DAQ controller preprocesses the sensor data and arrives at a decision based on the trained AI model. The DAQ controller feeds the decision output back to the robot controller via its digital-analogue I/O. In the branch of this local control loop, the preprocessed data is sent to the cloud in a suitable protocol with data encryption.

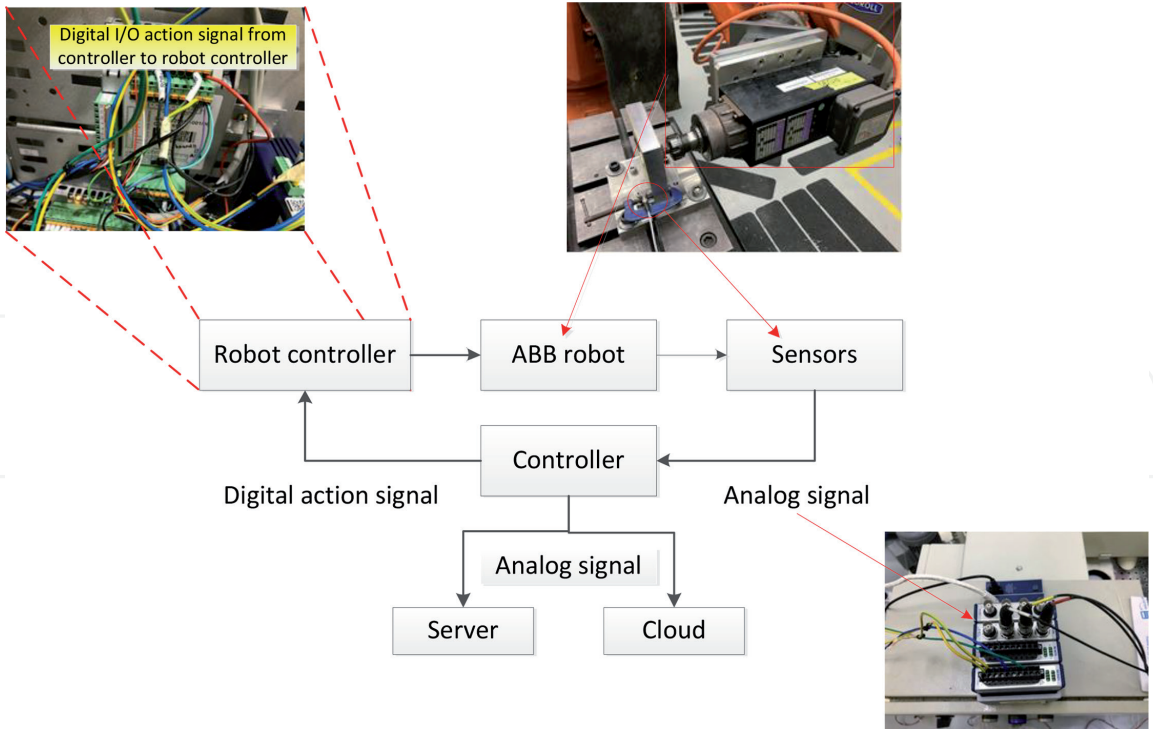


Figure 6.
Data path loop and DAQ to robot communication in deburring.

The implementation of data analysis to the cloud requires conversion of the machine learning algorithm to the cloud. Before moving into this, some fundamentals of the cloud services for Industry 4.0 implementation are explained to understand which platform is suitable for certain application.

4.2 Belt grinding

The current industrial practice of removing weld seam in manufacturing industry involves skilled operator using a belt sander. The component, when removed from the production for such operation, poses delay to the production cycle; fixturing, clamping, and unclamping cause a loss to the production volume. As an alternative, the belt grinder is integrated with the robot (in our case ABB6640). The tool path is programmed in the robot controller in such a way that weld seam and surrounding areas are blended. Machine vision-based solution is opted to analyse the state of the weld seam. If the weld seam is yet to be removed, the controller is triggered via flag setting through input/output pin. If the weld is removed, the controller is not triggered, and the completion message is sent.

After completion of each pass, the IP camera is triggered to capture the image of the component. The edge PC that triggers image capturing knows the state of tool path as it is the constant communication with the robot controller via TCP/IP. The image acquired is then passed through a secured network on MQTT protocol to a MATLAB instance running on a virtual machine inside Azure cloud services. The MATLAB instance has the deep learning model-based encoder-decoder architecture on predicting the pixel-wise state of the weld seam. The result is sent back to the edge PC, which initiates the tool path again. The cycle continues as long as the weld seam is completely blended with the surrounding surface. The deep learning algorithm is trained using four different weld seam states. The edge PC triggers the robot controller to execute the tool path unless weld seam state four is reached, i.e. where the weld seam is wholly removed. The schematic flow of data connectivity and decision-making is shown in **Figure 7**.

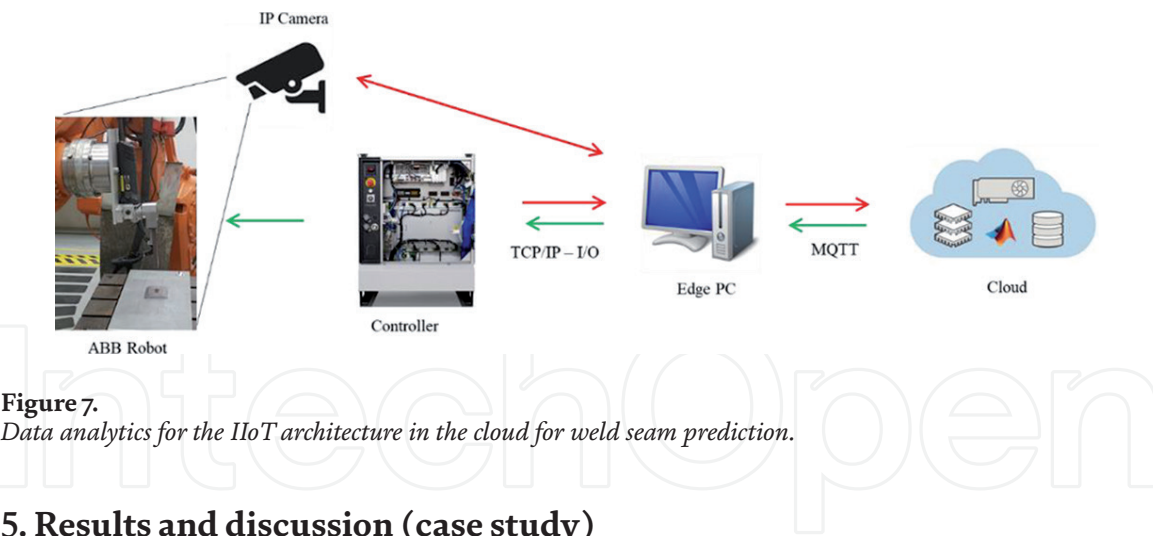


Figure 7.
Data analytics for the IIoT architecture in the cloud for weld seam prediction.

5. Results and discussion (case study)

5.1 Deburring

In order to establish data collection and analysis through machine learning to predict the deburred surface quality, several equipment have to be prepared. **Table 3** lists the systems used in this paper. Meanwhile, the system configuration for deburring is described in **Figure 8**. In **Figure 8**, it can be seen that the controller is the core equipment whose functions are to collect the sensor data, analyse, control robot action, and upload the data to the server and cloud. Thus, DAQ controller specification is crucial to determine the performance of this framework adaptation for deburring in the production line.

After understanding the data collection, the next step is to understand how to implement machine learning model into deburring process. A schematic is made to understand how the machine learning model processes the sensor data in deburring and predicts the physical features such as chamfer length and surface finish.

Figure 9 is the training phase to generate a model that correlates the input sensor data in deburring to the physical features measured on deburring work coupon as output. After the trained model satisfies the accuracy and repeatability required, the trained model is again implemented between the input and output. In the author’s previous journal paper, an example is discussed for the use of Welch’s estimate to compute power spectral density (PSD) to classify the number of passes and classify the vibration signal generated by the spindle based on adaptive neuro-fuzzy inference system (ANFIS) [18]. Meanwhile, in the second journal publication of this topic, the authors presented the results of fuzzy inference system (FIS) machine learning method to obtain the corresponding output as predicted surface finish quality of boss hole chamfer length and also the stage classification of deburring process [19]. A detailed result on feature extraction and machine learning

Equipment	Function
ABB robot	Robot for machining
DAQ Controller	Sensor data acquisition, analysis, and data transmission to the Internet
Sensors	Collecting the variables required from the process for analysis
Display	Displaying the sensors data, analysis process, and result

Table 3.
System for industry 4.0 implementation.

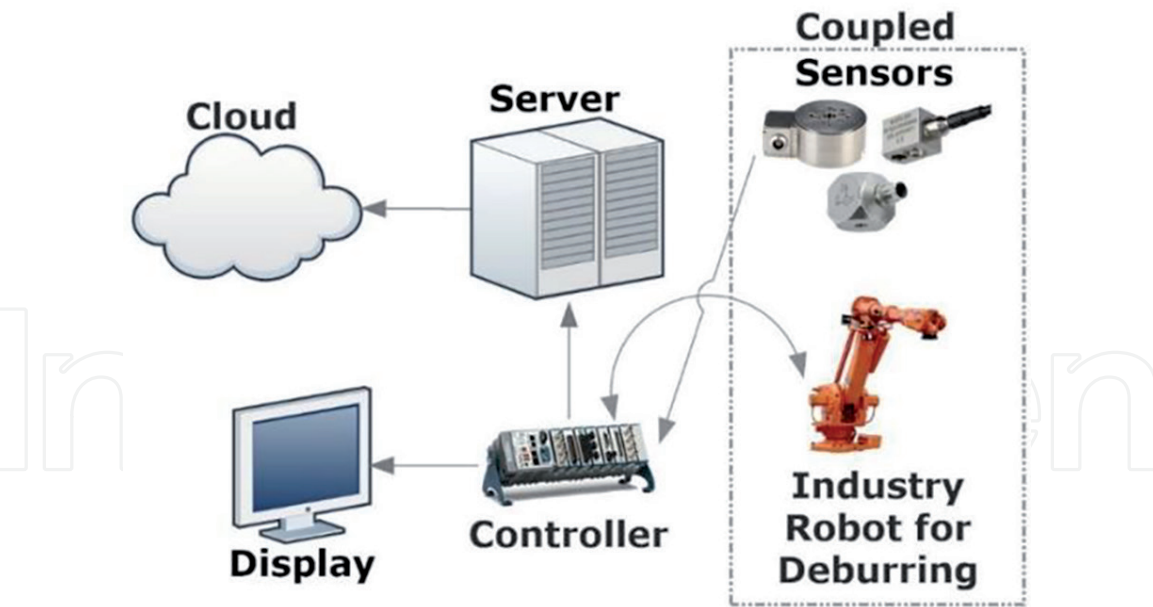


Figure 8.
Deburring equipment configuration and data path.

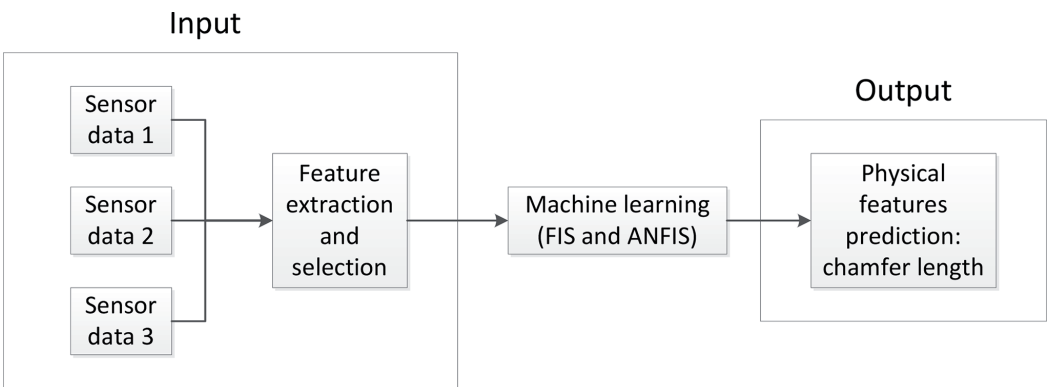


Figure 9.
Fuzzy inference system analysis model training on deburring.

method using ANFIS and FIS can be obtained from previous publications [18, 19]. A preliminary study of cloud computing to predict the output quality of deburring is presented in [19].

5.2 Belt grinding

An IP camera is introduced to capture the images of various stages of the weld seam. The IP camera is incorporated with the help of a tripod stand adjacent to the belt grinder. Surface images are captured at the end of every pass of robot arm across the weld seam (**Figure 10**). The camera system is capable of capturing and storing the surface images at a resolution of 1240 × 960 pixels. The images are subsequently labelled and sent to the Azure cloud with MATLAB Environment where the actual model training takes place. The variable grinding parameter used to remove weld seam is shown in **Table 4**.

Azure’s in-built architecture incorporates best practices for creating a full MATLAB desktop experience on Azure. This includes connecting to Azure from your local desktop using Remote Desktop Protocol (RDP). It sets up a single virtual machine containing MATLAB, a private virtual network with an Internet gateway, a private subnet, and a security group that opens the appropriate ports

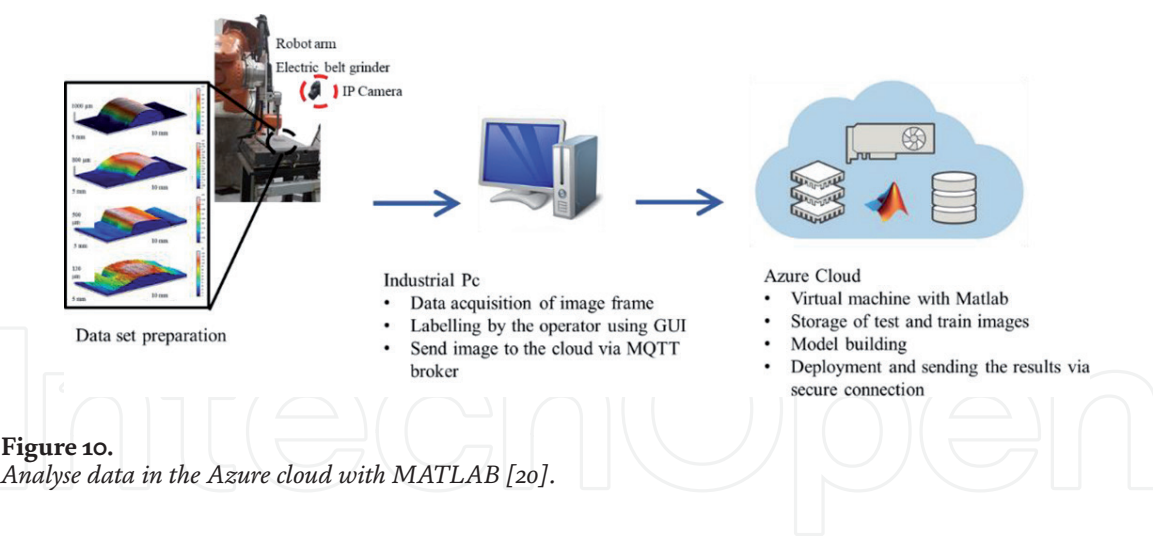


Figure 10.
Analyze data in the Azure cloud with MATLAB [20].

Parameter	Description
Belt grinding speed	5000–1100 RPM
Contact wheel diameter	10 mm, 24 mm
Hardness of contact wheel (polyurethane)	Shore A Hardness 30, 60, 90
Lubrication	Dry condition
Feed	10–40 mm
Belt finishing duration	Variable time
Operational mode	Position control

Table 4.
Parameters used in the belt grinding experimental trials.

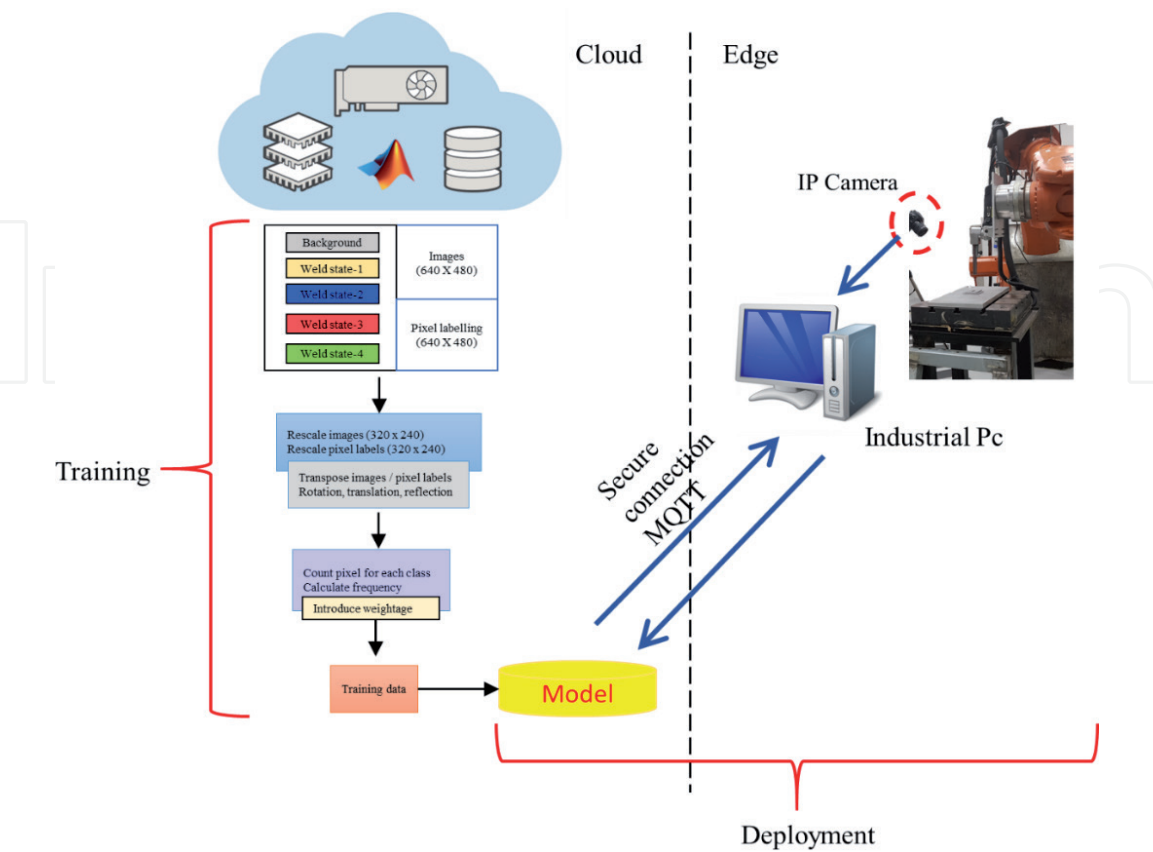


Figure 11.
General description of the proposed methodology to train and deploy the model in azure cloud.

for SSH and RDP access. For training and deploying, we will be using Azure cloud service as IaaS.

Figure 11 shows in detail the implementation of encoder-decoder-based deep learning model in the cloud. In total, 2000 images taken using the IP vision system are labelled offline and stored in the cloud. The MATLAB instance is created in the cloud, and required resources for computing are generally chosen before the creation of VM on which the MATLAB instance runs. The deep learning model is defined to identify four weld seam states. The VGG-16 network is retrained to identify the weld seam states. Pixel label layer of the default VGG-16 network is replaced by the customised label layer that would identify the weld seam state and background of an image. Distribution of pixel count for four different belt states and background is identified, and the corresponding weight is redefined on the final layer of the VGG-16 network. The weld state identification was performed in MATLAB deep learning toolbox. The augmented training image set is used for training, and it is ensured that the training accuracy increases and training loss decreases with the iteration count. The training is terminated once the parametric conditions are met. Once the model is ready for deployment, IP camera sends the image frame at an interval of the 1-second interval through secure MQTT to the cloud during the actual belt grinding trials. The script in the cloud passes the image into the developed model to make the pixel-wise classification. It is to be noted that the transfer speed of images from edge to cloud depends on the network traffic. The pixels' classified image is sent back to the edge PC for visualisation through MQTT secure network.

6. Conclusions

Two case studies related to the machine learning, deep learning, and cloud computing application towards smart manufacturing has been presented in this work. A methodology for integrating brownfield systems into IIoT framework to facilitate industry 4.0 adaptation is demonstrated. A simple data flow pipeline has been established between the edge and cloud framework via MQTT protocol. Cloud framework based on IaaS is used to deploy the deep learning model. However, it is to be noted that other protocols similar to MQTT can also be used for data transfer. The deployment of a robust analytical architectural framework cannot be just restricted to putting data processing and analysis software in place at cloud and edge. As a plant expands, the analytical framework and supporting hardware need to evolve organically. The framework should also support repeated installations and setup procedures being carried out simultaneously for brownfield equipment as well as systems with in-built IoT. Caution should also be exercised while amalgamating the analytical solution in cloud for existing legacy systems, in order to preserve their security and integrity.

Conflict of interest

The authors declare no conflict of interest.

IntechOpen

Author details


Vigneashwara Pandiyan¹ and Wahyu Caesarendra^{2*}

1 Smart Manufacturing Group, A*star Advanced Remanufacturing Technology Center, Singapore

2 Faculty of Integrated Technologies, Universiti Brunei Darussalam, Brunei Darussalam

*Address all correspondence to: wahyu.caesarendra@ubd.edu.bn

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. Distributed under the terms of the Creative Commons Attribution - NonCommercial 4.0 License (<https://creativecommons.org/licenses/by-nc/4.0/>), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited. 

References

- [1] Shariatzadeh N, Lundholm T, Lindberg L, Sivard G. Integration of digital factory with smart factory based on internet of things. *Procedia CIRP*. 2016;**50**:512-517
- [2] Jeschke S, Brecher C, Meisen T, Özdemir D, Eschert T. Industrial internet of things and cyber manufacturing systems. In: *Industrial Internet of Things*. Cham, Switzerland: Springer; 2017. pp. 3-19
- [3] Zhou K, Liu T, Zhou L. Industry 4.0: Towards future industrial opportunities and challenges. In: *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, IEEE. 2015. pp. 2147-2152
- [4] Chen Y. Integrated and intelligent manufacturing: Perspectives and enablers. *Engineering*. 2017;**3**(5):588-595
- [5] Zhong RY, Xu X, Klotz E, Newman ST. Intelligent manufacturing in the context of industry 4.0: A review. *Engineering*. 2017;**3**(5):616-630
- [6] Stojanovic L. Intelligent edge processing. In: *Machine Learning for Cyber Physical Systems*. Berlin, Heidelberg: Springer; 2020. pp. 35-42
- [7] Domingo Galindo L. The Challenges of Logistics 4.0 for the Supply Chain Management and the Information Technology. Norway: NTNU; 2016
- [8] Chen X, Limchimchol T. Monitoring grinding wheel redress-life using support vector machines. *International Journal of Automation and Computing*. 2006;**3**(1):56-62
- [9] Pandiyan V, Caesarendra W, Tjahjowidodo T, Tan HH. In-process tool condition monitoring in compliant abrasive belt grinding process using support vector machine and genetic algorithm. *Journal of Manufacturing Processes*. 2018;**31**:199-213
- [10] Zheng S, Ristovski K, Farahat A, Gupta C. Long short-term memory network for remaining useful life estimation. In: *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*, IEEE. 2017. pp. 88-95
- [11] Jiang J-R. An improved cyber-physical systems architecture for industry 4.0 smart factories. *Advances in Mechanical Engineering*. 2018;**10**(6):1687814018784192
- [12] Xu X. From cloud computing to cloud manufacturing. *Robotics and Computer-Integrated Manufacturing*. 2012;**28**(1):75-86
- [13] Tao F, Cheng Y, Da Xu L, Zhang L, Li BH. CCIoT-CMfg: Cloud computing and internet of things-based cloud manufacturing service system. *IEEE Transactions on Industrial Informatics*. 2014;**10**(2):1435-1442
- [14] Bhardwaj S, Jain L, Jain S. Cloud computing: A study of infrastructure as a service (IAAS). *International Journal of Engineering and Information Technology*. 2010;**2**:60-63
- [15] Peng J, Zhang X, Lei Z, Zhang B, Zhang W, Li Q. Comparison of several cloud computing platforms. In: *2009 Second International Symposium on Information Science and Engineering*. 2009. pp. 23-27
- [16] Ren X, Cabaravdic M, Zhang X, Kuhlenkötter B. A local process model for simulation of robotic belt grinding. *International Journal of Machine Tools and Manufacture*. 2007;**47**(6):962-970
- [17] Ren X, Kuhlenkötter B. Real-time simulation and visualization of robotic belt grinding processes. The

International Journal of Advanced
Manufacturing Technology.
2008;35(11-12):1090-1099

[18] Caesarendra W, Wijaya T,
Tjahjowidodo T, Pappachan BK,
Wee A, Roslan MI. Adaptive neuro-
fuzzy inference system for deburring
stage classification and prediction for
indirect quality monitoring. *Applied
Soft Computing*. 2018;72:565-578

[19] Caesarendra W, Pappachan BK,
Wijaya T, Lee D, Tjahjowidodo T,
Then D, et al. An AWS machine
learning-based indirect monitoring
method for deburring in aerospace
industries towards industry 4.0. *Applied
Sciences*. 2018;8(11):2165

[20] MATLAB and MATLAB in the
Cloud. Natick, Massachusetts, United
States: The MathWorks, Inc.; 2018b