Ref. Ares(2024)2354004 - 29/03/2024

D5.1 – Industrial Equipment Use Suite v1

WP5 – BUILD: AIDEAS 4 Industrial Equipment Use

Co-funded by the European Union

Document Information

GRANT AGREEMENT NUMBER	101057294	ACRONYM	AIDEAS
FULL TITLE	AI Driven Industrial Equipment Product Life Cycle Boosting Agility, Sustainability and Resilience		
START DATE	01-10-2021	DURATION	36 months
PROJECT URL	https://aideas-project.eu/		
DELIVERABLE	D5.1 – Industrial Equipment Use Suite v1		
WORK PACKAGE	WP5 – BUILD: AIDEAS 4 Industrial Equipment Use		
DATE OF DELIVERY	CONTRACTUAL 31-Mar-2024	ACTUAL	29-Mar-2024
NATURE	Other	DISSEMINATION LEVEL	Public
LEAD BENEFICIARY	IKER		
RESPONSIBLE AUTHOR	Ana Gómez		
CONTRIBUTIONS FROM	1-CERTH, 3-UNINOVA, 4-IKER, 7-ITI, 10-XLAB		
TARGET AUDIENCE	1) AIDEAS Project partners; 2) industrial sector; 3) other European projects (H2020/HORIZON EU); 4) scientific community.		
DELIVERABLE CONTEXT/ DEPENDENCIES	This document is the first iteration of the Industrial Equipment Use Suite. Its relationship to other documents is as follows: - D5.2: Industrial Equipment Use Suite v2 (M24). - D5.3: Industrial Equipment Use Suite v3 (M36).		
EXTERNAL ANNEXES/ SUPPORTING DOCUMENTS	None		
READING NOTES	None		
ABSTRACT	D5.1 aims to gather a description of all solutions developed inside WP5, i.e. related to the Use lifecycle phase of Industrial Equipment. This description includes an overview, features, technical specifications and the implementation status. Five toolkits are included in this suite: Machine Calibrator, Condition Evaluator, Anomaly Detector, Adaptive Controller and Quality Assurance.		

Document History

VERSION	ISSUE DATE	STAGE	DESCRIPTION	CONTRIBUTOR
0.1	14-Mar-2024	ToC	Table of Contents	IKER
0.2	06-Mar-2024	Draft	1st Draft with initial contributions from all partners	UNI, IKER, XLAB, ITI, CERTH
0.3	12-Mar-2024	Review	Conduction of Internal Reviews and Plagiarism check	CeSI, UNIVPM, UPV
0.4	15-Mar-2024	Draft	2nd Draft addressing the comments by the reviewers	IKER, UNI, ITI
1.0	29-Mar-2024	Final Doc	Quality check and issue of final document	CERTH

Disclaimer

Copyright message

TABLE OF CONTENTS

ABBREVIATIONS/ACRONYMS

AC	Adaptive Controller
AI	Artificial Intelligence
AD	Anomaly Detector
API	Application Programming Interface
CE	Condition Evaluator
CNC	Computer Numerical Control
CPU	Central Processing Unit
CSV	Comma Separated Value
DB	DataBase
GDPR	General Data Protection Regulation
GPU	Graphics Processing Unit
IPR	Intellectual Property Rights
JSON	JavaScript Object Notation
MC	Machine Calibrator
ML	Machine Learning
MP	Machine Passport
NN	Neural Networks
ORM	Object-Relational Mapping
PWDS	Partial Wall Thickness Control
QA	Quality Assurance
REST	Representational State Transfer
RPM	Revolutions Per Minute
SQL	Structured Query Language
SVM	Support Vector Machine
UI	User Interface
XAI	Explainable Artificial Intelligence
DSS	Decision Support System
UI	User Interface

Executive summary

The objective of AIDEAS will be to develop AI technologies that support the complete lifecycle of industrial equipment, encompassing design, manufacturing, use, and repair/reuse/recycle processes, developing for these four different Suites, one for each lifecycle phase. The primary objective is to enhance European machinery manufacturing companies' sustainability, agility, and resilience.

This deliverable focuses on the description of the solutions developed inside the Use suite that centres around AI technologies that offer added value to industrial equipment users. These technologies provide enhanced support for installation, initial calibration, production processes, and quality assurance, ensuring optimal operating conditions.

The description of each solution or toolkit includes:

- A first overview for non-technical users.
- The solution's features and functionalities, including the problems that each solution or toolkit solves.
- Its technical specifications with the technologies used and referring to any background being used.
- The implementation status at M18 and the future development plans.

The six solutions included in this suite are:

- AIDEAS Machine Calibrator.
- AIDEAS Condition Evaluator.
- AIDEAS Anomaly Detector.
- AIDEAS Adaptive Controller.
- AIDEAS Quality Assurance.
- AIDEAS Machine Passport – Use.

Some information regarding the solutions can also be found in other deliverables from WP2, namely D2.2, where the functionality, usage and implementation of the solutions is detailed. The reader can also refer to the web documentation located on the AIDEAS website https://viewpoints.aideas-srv.cigip.upv.es/.

All the solutions from this Suite can be found in the AIDEAS GITLAB repository located in https://gitlab-cigip.alc.upv.es/aideas.

Document structure

Each section of the document contains the information of one of the solutions, with a final section for conclusions, namely:

Section 1: Machine Calibrator.

Section 2: Condition Evaluator.

Section 3: Anomaly Detector.

Section 4: Adaptive Controller.

Section 5: Quality Assurance.

Section 6: Machine Passport.

Section 7: Conclusion.

1. Machine Calibrator

1.1 Overview

The process of configuring and calibrating new industrial machines poses a challenge in standardization due to variations introduced by different manufacturers, diverse specifications, client-specific adaptations, and inherent distinctions across application fields.

To address this challenge, the AIDEAS AI^{MC} solution aims to assist machine installers and operators in the initial calibration and configuration of new machines, considering the unique requirements of each customer and factory through the application of AI techniques.

The AIDEAS Machine Calibrator AI undergoes training using the parameterizations acquired during the commissioning phase, manufacturing, and initial calibration of industrial machines. This training occurs at the time of installing a new machine, capturing the intricacies of each setup.

The solution aims to learn from operational information from experienced users, calibration parameters can be optimised and adjusted to effectively meet specific process needs.

1.2 Features

AI^{MC} intends to develop a solution that can assist the machine installers and operators in the initial calibration and configuration of new machines.

The installation of industrial machines can take many days, due to its dimensions and the need for precision alignments, meaning that small deviations impact the final product with unconformities to the measurements requested by the customer. The solution should be used to support the machine installer to make calibration faster and more accurate.

The quality of the articles produced can often be checked hours or days after production, making the correction of the calibration errors a very time-consuming delay, that implicates the reconfiguration of the machine parameters and further iteration of testing. The solution should analyse the previously available historical data in order to predict the corrections of the parameters that deviate from the conditions of the production process.

The calibration process aims to align the machine frame and base involving a trial-and-error method, requiring qualified installers to adjust the axial frame and level the machine. The introduction of a machine vision system allows for early detection of off-axis components, enabling instant corrections and significantly reducing overall calibration time.

During the calibration process, a camera will be used to visualize the movements of the moving parts, in real-time, and will be accompanied by a solution that will evaluate the calibration and indicate if an error is identified. This process starts when the technician finishes installing the machine and starts its calibration, thus executing the AI^{MC} which will activate the camera and start the data capture, then the machine installer chooses an axis and defines a travel path that will be evaluated to check if there is an error in the movement. After this evaluation, it indicates to the machine installer what the error is so that the machine can be re-calibrated. This process will be carried out for all axes and then repeated to check if any displacement occurred during the calibration.

Configuration processes can be optimized based on various objectives, resulting in variable production times and different material and energy consumption/waste. The automatic parameterization of each production program, adapted to specific objectives, simplifies the operator's workload and minimizes the need for frequent recalibration procedures.

During the calibration process, several critical steps are undertaken to ensure the optimal functioning of industrial machines. The calibration includes a thorough check of the machine mechanics and the configuration of production parameters. Program optimization is then carried out, covering fluid flow speed and pressure profile programming, measurement and adjustment of wall thickness, circular material distribution setting, weight/mass of the article setting, material temperature configuration, and extrusion head temperature setting.

Coextrusion is checked and optimized by programming the extruder RPM, and customer specifications are validated to meet their requirements. Various machine parameters are configurable during calibration, including water temperature return, voltage heating, apparent current heating, active power heating, blowing pressure, clamping force, cycle duration, extruder RPM and current, extruder mass temperature and pressure, different temperature zones, wagon torque, mold torque, blowing pin torque, transport wagon and blowing pin motion control and cutting unit operation.

Quality control results are monitored, covering tightness, spout ovality, gross and net weight, wall thickness, and energy consumption. Additionally, the calibration process involves cycle time optimization by fine-tuning the transport wagon's motion, synchronizing it with the blowing pin. The cooling time inside the mold is also optimized, finding a balance between faster cooling times and minimizing material/energy consumption and waste. This comprehensive calibration approach ensures the efficient and precise operation of the industrial machines.

In result of this study, the main features and functionalities offered by the AIDEAS Machine Calibrator are the following:

- **Import data:** machine data will be imported in CSV, G-code, JSON or video format, depending on the use case to be used.
- **Data validation and preprocessing:** Validates training data and ensures input data is in the correct format before feeding it into the model.
- **Machine Calibration:** allows configuration indicating the type of machine and what type of material will be used in its production.
- **Create and export models:** provide the trained algorithm for use on other machines and identify whether the system is behaving as expected.
- **Obtain and display results:** depending on the data obtained, the algorithm gives instructions to the machine installer or operator to help them to configure the machine to obtain the desired calibration. The model will be in operation until it reaches the ideal calibration, adjusting the results according to the technicians' modifications and making real-time adjustments.
- **Export Data:** Provide the data to Machine Passport and makes it easy for the user to export it directly if necessary.

1.3 Technical specifications

The backend of AIDEAS Machine Calibrator was developed from scratch, no previous developments were integrated into the solution, while the front-end of the solution was developed using the UPV model as a starting point (in REACT).

The AIDEAS Machine Calibrator will be fed from two different sources, depending on which use case it will be used in, in one situation one of the sources will provide a CSV file with machine data, such as temperature, and position, among others. On the other hand, a G-code and video files will be provided, the G-code file will contain the spindle positions and the video file will contain the real-time movement of the spindle.

AI^{MC} will feature a supervised learning approach based on operational information from experienced users, calibration parameters can be optimized and adjusted to effectively meet specific process needs. Standard machine learning will use TensorFlow, PyTorch, and scikit-learn as the development tools. For the image processing, the AI^{MC} solution uses the anomalib library with PyTorch which can be deployed locally, having the image data passed for processing on the locally accessible mounted volume.

The AI^{MC} backend is developed using Python and Javascript (Node.js) as the framework for the API server. The backend provides the API endpoints that the frontend can communicate with, send requests, and get the results. The backend will also connect the frontend with the PostgreSQL database being used.

For deployment, docker is used because it is the most used containerization solution and because of our experience with it. Docker also makes it easy to deploy the packaged application to the runtime environment and is widely supported by deployment tools and technologies.

1.4 Implementation status

1.4.1 Current implementation

The current development of the AI^{MC} solution from the Stone Cutting Machine is pending the definition of the machine parameters used during the normal calibration process, examples of the available machine data (such as a template G-code file for standardized axis movement, default machine configuration parameters and operational data). The AI model for this solution requires that the aforementioned data be made available to begin its development.

For the Injection Blow Machine, a dataset of machine parameters was made available, containing temperatures, extrusion, pressure and power measurements of their machine. This dataset requires annotation and classification of the parameters that influence the article quality and production speed.

The current version of the AI^{MC} solution can be found in https://gitlab-cigip.alc.upv.es/aideas/industrial-equipment-use/mc-machine-calibrator/mc.

1.4.2 Next developments

The AI^{MC} requires the collection of additional datasets and refinement of the datasets to advance the development of the AI models.

Furthermore, the following tasks are planned to be carried out:

- Connection with the machine passport.
- Import the CSV file into the solutions.
- Connect the camera with the solution for the 5 axes calibration.
- Conclude the development of the CNC calibration algorithm based on image.
- Create the application to show the results.
- Install the solutions in the pilot’s premises.
- Assess the impact with the pilot’s and refine the solutions.

2. Condition Evaluator

2.1 Overview

The primary aim of condition evaluation, or monitoring, is to act as an early warning system for detecting potential issues or malfunctions in systems that may not be immediately visible at first sight. Condition monitoring plays a crucial role in effectively managing assets in two ways: firstly, by monitoring the behaviour of equipment's process variables through sensors, and secondly, by identifying the characteristics and root causes of faults if they occur.

To address this issue and as the complexity of equipment continues to rise rapidly, the AIDEAS Condition Evaluator (AI^{CE}) will enhance productivity, minimize downtimes, and ensure the safety and reliability of installations.

The AI^{CE} is a toolkit for determining the condition of the machine as a whole or of some of its components when it is in working conditions in the factory where it is being used.

2.2 Features

The main features and functionalities offered by the AI^{CE} are the following:

- **Import Data**: the user can select data from different sources, such as: Databases (e.g. MongoDB), CSV or EXCEL files.
- **Data Validation and Preprocessing:** validates the training data and ensures that the input data is in the correct format before feeding it into the model.
- **Machine Configurator**: allows defining the current machine configuration in order to identify its different component and its associated variables. It also allows to define which is to be considered as "normal" behaviour.
- **Create and Export Models**: provides different algorithms, which train on the available pre-processed data to determine if the system is working properly, or if the system is not behaving as expected.
- **Obtain Predictions and Display Results**: determines if the system is behaving as expected or if it is deviating from normal behaviour, giving a score, with new data using the trained models. When a deviation is observed this can be used as a trigger for the AIDEAS Prescriptive Maintenance solution, which will evaluate the Remaining Useful Life of the component.
- **Export Data**: sends the data to the Machine Passport and facilitates the user to export it directly if needed.

The main problem that can be solved by using this solution is the identification of deviations from normal behaviour, analysing any repetitive operation the machine (or one of its components) performs periodically. With this approach the user will be able to, for example:

- Monitor energy consumption.
- Identify increasing vibrations.
- Changes in temperature patterns

2.3 Technical specifications

The AI^{CE} backend has been developed from scratch, based on new concepts, using previous knowledge and expertise in the field, and considering current state of the art from the literature. While solution's frontend has been developed using UPV's template as the starting point. No previous developments have been integrated into the solution.

The AI^{CE} is fed with time series data in which there are several process variables distributed in columns while every row represents a single timestamp. For the condition evaluator it is also needed to define where the normal behaviour and subsequent datasets to compare with are happening during machine's normal operation, condition evaluation is performed under specific cycles, tasks that repeat over time. In addition to raw data, a machine configuration file is needed to give context to the evaluation performed, to know exactly in which component have an effect.

The AI^{CE} characterizes the normal behaviour during the test cycles mentioned above using statistical methods for a set of interest variables. These characterized patterns will then be used and compared to future test cycles to evaluate how the machine is performing, compared to the characterized pattern. For the moment, and because of the amount of data received from pilots, no more complex approaches have been developed. The model files can be saved, under MinIO storage, to the backend of the solutions as a pkl file to be used with new data. At the moment, solution's outputs are only displayed in the UI.

The backend of the AI^{CE} is developed using python and FLASK as the framework for the API server. The backend provides the API endpoints with which the frontend can communicate to, send requests, and obtain the results.

The frontend of the solution is developed in REACT.

For deployment, docker is used since it is the most widely used containerization solution and due to our expertise with it. Docker also makes it easy to deploy the packaged application into the runtime environment and is widely supported by deployment tools and technologies.

For internal storage a MinIO server is used. MinIO is a High-Performance Object Storage.

2.4 Implementation status

2.4.1 Current implementation

Currently, the solution includes the following features:

- Creation or upload of the machine configuration file. The file can be saved too.
- Upload and visualize data from csv or excel files. Data can be visualized in both table and graphs.
- MongoDB connection and data visualization. The solution can be connected to an external Mongo database and visualize its data in both table and graphs.
- Creation of pkl model files. Models can be created choosing the input data, the set of variables to use and the test cycle to study. Training results can be visualized as a report or in graphs, showing the current machine condition. The model can be saved too.

- Evaluating the condition given a model and a new dataset, results can be visualized as report, table and graphs.

The current version of the AI^CE solution can be found in https://gitlab-cigip.alc.upv.es/aideas/industrial-equipment-use/ce-condition-evaluator/ce.

2.4.2 Next developments

Future developments will include:

- Communication with the machine passport, sending the obtained results.
- Increase the number of algorithms to choose from during training phase. Adding more complex algorithms than the ones presented and deep learning algorithms.
- Obtain results in real time or based in a predefined time interval.
- Introduce user management and different accesses depending on the logged user using keycloak. Integrate these users under MinIO so each user can access certain information.
- Provide a version of the tool far less customizable, e.g. targeted to unexperienced users, with predefined parametrizations and/or with less functionalities.
- Enhancement of plots when huge datasets are loaded.
- Keep a history of the results, saved under an internal database, for example MongoDB.

3. Anomaly Detector

3.1 Overview

An anomaly, also called an outlier, is traditionally defined as an observation that significantly deviates from others. Identifying anomalies can be advantageous for many industries, as they could signal production failures, defects, undesirable events, or machinery degradation, all of which are critical for optimizing equipment availability. Specifically, time series anomaly detection analyses such occurrences over time.

Applying AIDEAS Anomaly Detector (AI^{AD}) in different manufacturing processes can prevent the appearance of defective parts, enabling its reuse or rejection, as necessary, providing benefits in the production workflow.

The AI^{AD} is a toolkit for detecting anomalies at component-level or of the machine as a whole when it is in working conditions in the factory where it is being used.

3.2 Features

The main features and functionalities offered by the AI^{AD} are the following:

- **Import Data**: the user can select data from different sources, such as: Databases (e.g. MongoDB), CSV or EXCEL files.
- **Data Validation and Preprocessing:** validates the training data and ensures that the input data is in the correct format before feeding it into the model.
- **Machine Configurator**: allows defining the current machine configuration in order to identify its different component and its associated variables.
- **Create and Export Models**: provides different algorithms, which train on the available pre-processed data to determine if there is an anomaly in an unsupervised way.
- **Obtain and Display Results**: using trained models and new data it is determined if there is an anomaly or if the system is behaving as expected.
- **Export Data**: sends the data to the Machine Passport and facilitates the user to export it directly if needed.

The main problem that can be solved by using this solution is the identification of outliers, analyzing any or several machine signals detecting anomalies during machine operation.

3.3 Technical specifications

The AI^{AD} backend has been developed from scratch, based on new concepts, using previous knowledge and expertise in the field, and considering current state of the art from the literature. While solution's frontend has been developed using UPV's template as the starting point. Some frontend components developed in i4Q project, in task 4.6 i4Q Digital Twin, have been reutilized or used as inspiration, for example the model parametrization component.

The solution is fed with time series data in which there are several process variables distributed in columns while every row represents a single timestamp. In addition to raw data, a machine configuration file is needed to give context to the anomalies detected, to know exactly in which component have an effect.

The AI algorithms for anomaly detection have been selected from the scikit learn library. Unsupervised learning methods are used such as Nearest Neighbors, K-means and one class SVM. Also, other algorithms such as isolation forest or autoencoders have been tested but have not been implemented on the solution yet. The model files can be saved, under MinIO storage, to the backend of the solutions as a pkl file to be used with new data. At the moment, solution's outputs are only displayed in the UI.

The backend of the AI^{AD} is developed using python and FLASK as the framework for the API server. The backend provides the API endpoints with which the frontend can communicate to, send requests, and obtain the results.

The frontend of the solution is developed in REACT.

For deployment, docker is used since it is the most widely used containerization solution and due to our expertise with it. Docker also makes it easy to deploy the packaged application into the runtime environment and is widely supported by deployment tools and technologies.

For internal storage a MinIO server is used. MinIO is a High-Performance Object Storage

3.4 Implementation status

3.4.1 Current implementation

State of development that has the solution in the March 2024 delivery.

Currently, the solution includes the following features:

- Creation or upload of the machine configuration file. The file can be saved too.
- Upload and visualize data from csv or excel files. Data can be visualized in both table and graphs.
- MongoDB connection and data visualization. The solution can be connected to an external Mongo database and visualize its data in both table and graphs.
- Creation of pkl model files. Models can be created choosing the input data, the set of variables to use, the algorithm and its parameters. Training results can be visualized as a report or in graphs, showing the detected anomalies. The model can be saved too.
- Obtaining anomalies given a model and a new dataset, results can be visualized as report and graphs.

The current version of the AI^{AD} solution can be found in https://gitlab-cigip.alc.upv.es/aideas/industrial-equipment-use/ce-condition-evaluator/ad.

3.4.2 Next developments

Future developments will include:

- Communication with the machine passport, sending the list of anomalies detected.
- Increase the number of algorithms to choose from during training phase. Adding more complex algorithms that the ones presented and deep learning algorithms.
- Obtain results in real time or based in a predefined time interval.

- Introduce user management and different accesses depending on the logged user using keycloak. Integrate these users under MinIO so each user can access certain information.
- Provide a version of the tool far less customizable, e.g. for unexperienced users, with predefined parametrizations and/or with less functionalities.
- Enhancement of plots when huge datasets are loaded.
- Keep a history of the results, saved under an internal database, for example MongoDB.
- Show the results in table format and give the possibility to export them in different formats, .xls, .csv, .pdf.

4. Adaptive Controller

4.1 Overview

Nowadays, industrial machines manufacturers face increasingly challenging specifications as, for the best economy of their operation, the machine should have as high accuracy and be as fast as possible, to ensure an optimal and efficient solution.

If the machine cannot meet with the client requirements, this inevitably will lead to higher costs and longer lead-times, and perhaps the possibility to lose the client. However, providing a fully functional machine that satisfies every design or operation requirement, will often result in high productivity and growth for businesses due to the technology improvement.

In this line, the AI-Adaptive Controller (AI^{AC}) solution enhances machines with intelligence and autonomy, minimizing human errors by automating complex tasks. Using field data, AI^{AC} develops a twin digital model of the machine, to fine-tune actual controllers to optimize the machine process.

4.2 Features

The main features and functionalities that the AI^{AC} solution offers are the following:

- **Import Data:** this feature allows the user to read data from different sources, such as MongoDB or Excel Files.
- **Data Validating and Preprocessing:** this feature validates the training data imported by the user and ensures that the data is in correct format before feeding it into the model.
- **Machine Configurator:** this feature allows to define the machine configuration, selecting the desired components and associated variables.
- **Create and Export Models:** this feature allows the user to choose between different model training algorithms, to choose the most appropriate to obtain an accurate relationship of the actual inputs/outputs.
- **Create and Export Controllers:** this feature provides an adaptive controller algorithm, which trains user-defined models and parameters, controls the systems and optimizes them.
- **Obtain Controller Evaluation:** this feature analyses and evaluates the controller's performance.
- **Export Data:** this feature sends the data to the Machine Passport and facilitates the user the possibility to export it directly.

This tool is designed to simplify the calculation of the optimal input for a machine by employing an adaptive controller, which adjusts parameters based on the machine's operating envelope.

4.3 Technical specifications

The AI^{AC} solution is fed with data that provides information about the status of the machine. Among this information we can find specific machine’s associated variables, such as control inputs or internal states. This information is needed to determine whether the machine control inputs need to be modified due to its performance or if it is operating optimally.

The AI algorithms for the AI[AC] solution have been selected from the scikit learn library. Supervised learning methods such as NN, Gaussian Regression, Lasso Regression and Ridge Regression. For the adaptive controller a self-made Fuzzy Gaussian Controller has been designed, which is based on a user-defined input and output set and predicts an optimal output. The models files can be saved, under MinIO storage, to the backend of the solutions as a pkl file to be used with the new data. At the moment, solution's outputs are only displayed in the UI.

The backend of the AI[AC] is developed using python and FLASK as the framework for the API server. The backend provides the API endpoints with which the frontend can communicate to, send requests, and obtain the results.

The front end of the solution is developed in REACT, using UPV's AIDEAS template as the starting point of the development.

For deployment, docker is used since it is the most widely used containerization solution and due to our expertise with it. Docker also makes it easy to deploy the packaged application into the runtime environment and is widely supported by deployment tools and technologies.

For internal storage a MinIO server is used. MinIO is a High-Performance Object Storage.

4.4 Implementation status

4.4.1 Current implementation

Currently, the solution includes the following features:

- Creation or upload of the machine configuration file.
- Upload and visualize data from csv and excel files.
- Creation of pkl model files. Models can be trained choosing the inputs and outputs sets of variables. Showing, in terms of different evaluation methods, the effectiveness of the training. The model training results can be visualized in different plots.
- The adaptive controller has been designed in Python environment.

The current version of the AIDEAS Adaptive Controller solution can be found in https://gitlab-cigip.alc.upv.es/aideas/industrial-equipment-use/adaptive-controller/ac.

4.4.2 Next developments

Future development will include:

- Integration of the adaptive controller into the back-end environment.
- Communication with the machine passport, send the calculated optimal control.
- Include graphical visualization of the controller training.
- Introduce user management and different accesses depending on the logged user using keycloak. Integrate these users under MinIO so each user can access certain information.
- Provide a version of the tool far less customizable, e.g. for unexperienced users, with predefined parametrizations and/or with less functionalities.
- Enhancement of plots when huge datasets are loaded.
- Keep a history of the results, saved under an internal database, for example MongoDB.

5. Quality Assurance

5.1 Overview

The overall objective of the AI^{QA} solution is to detect and localize visual defects of the product taken in production based on computer vision techniques. For this purpose, two solutions are being developed: AI^{QA} -2D and AI^{QA} -3D.

The AI^{QA} -2D solution currently analyzes 2D images of the products and detects and classifies defects on them (stones in our pilot). Additionally, the solution produces an image showing a heatmap of defects from the 2D image and gives a ranking of the product.

Conversely, the primary objective of the AI^{QA} -3D solution is to employ computer vision for the detection, localization, and classification of 3D captures of products taken during production. Currently, it classifies between defective and correct products.

5.2 Features

The main features of both AI^{QA} -2D and AI^{QA} -3D solution are already implemented as follows:

- **Image Loading:** Enables users upload image data into the system.
- **Data Validation and preprocessing:** Validates training data imported by the user. Ensures data is in the correct format before feeding it into the model.
- **Import and configure models:** for AI^{QA} -2D the solution uses the most appropriate SOtA method for industrial 2D usage. The AI^{QA} -3D allows users to choose from different models thus facilitating the selection of the most appropriate algorithm for accurate relationships between inputs and outputs.
- **Export data:** for AI^{QA} -2D the solution returns a heat-map overlay and a defect-detection rank number as the output, provided the product image. The higher the rank, the more likely the product image has defects detected while the AI^{QA} -3D solution provides users with a convenient and efficient way to manage their data, displaying the quantity of both approved and defective products.

5.3 Technical specifications

Currently the AI^{QA} -2D solution uses anomalib library with PyTorch and can be deployed locally, the image data is passed in the locally accessible mounted volume. At this point the output images are stored in a specific output map along with its rank in a textual file format.

This solution will be deployed as a service in a Docker container utilizing GPU (default), or CPU resources. The application's AI lifecycle (e.g., experiment and model versioning) will be managed using MLFlow workflow. Image data will be primarily stored on a user's on premises resource which effectively bypasses any privacy or IPR issues related to data. The solution will, nevertheless, additionally support the possibility of storing image data in a cloud-based object such as datalakes. The image processing metadata will be stored primarily in a NoSQL database (such as MongoDB). Microsoft Azure Cloud will be used for the deployment of the solution. We do not plan specific development of user interfaces, as the main purpose of the solution is the integration into existing applications - either on the machine itself, or other applications that are currently used to process imagery data on the pilot side.

In future releases the plan is to implement an asynchronous RESTful API where JSON inputs and outputs will be defined, along with a DB storage for storing image and request metadata for classification. The solution provider will also provide a demo user interface probably based on Streamlit4 enable pilot-independent demonstration of the solution.

AI^{QA} -3D solution utilizes PyTorch and is available as a service for easy integration. It comes as a Docker container that can efficiently use GPU (default) or CPU resources. To address privacy and intellectual property concerns, user image data is stored on-premises, eliminating potential issues related to external data storage. Image processing metadata is primarily stored in a SQL database, such as SQLite. The user interface is developed in REACT, using UPV's AIDEAS template for simplicity and functionality. Additionally, the solution will offer a REST API interface developed in python for smooth integration into existing systems.

5.4 Implementation status

5.4.1 Current implementation

Currently the AI^{QA} -2D solution, is a locally deployed solution for testing using limited samples provided by pilot provider (Stone anomaly detection) and focusing mostly on model quality and data resilience. The current results are encouraging but there is still room for improvement.

Feature status is as follows:

- **Primary version of the user interface:** is not intended to be developed for production, but only for demo purposes, since the solution will be deployed as a RESTful API.
- **Upload 2D product images data to the solution:** Users can upload 2D images data to the solution – mounted volume currently.
- **Classification of the uploaded images:** The solution can currently classify stone product features for analysis and categorizing uploaded images between correct and defective.
- **Model's result visualization:** The solution presents a visual representation of the results generated by the selected model and method.

The AI^{QA} -3D solution currently includes the following features:

- **Primary version of the user interface:** The initial release features a foundational user interface, providing users with a basic yet functional interaction platform.
- **Upload 3D imagery data to the solution:** Users can upload 3D imagery data to the solution.
- **Classification of the uploaded images:** The solution includes a classification feature for analyzing and categorizing uploaded images between correct and defective.
- **Selection of available models:** Users can select from a range of available models within the solution.
- **Model's result visualization:** The solution presents a visual representation of the results generated by the selected model.

The current version of the AIDEAS Quality Assurance solution can be found in https://gitlab-cigip.alc.upv.es/aideas/industrial-equipment-use/qa-quality-assurance/qa.

5.4.2 Next developments

For the AI^{QA} -2D solution, the plan is to deploy the solution in a docker container which will utilize both GPU (default) and CPU-based resources. The solution's AI lifecycle (e.g., experiment and model versioning) will be managed using MLFlow2 (e.g., cloud-based Databrick's MLFlow). Data should be stored mainly on user premises (existing databases and disk resources), as data can be prone to different GDPR and privacy rules. The application will nevertheless additionally support a cloud-based object such as datalakes.

For the AI^{QA} -3D solution, future development will include:

- **Defect detection and localization:** The next phase of development aims to extend the existing classification capabilities. The focus will be incorporating advanced defect detection and localization functionalities, particularly on the 3D surfaces of incoming objects.
- **Expand models to other needs:** As part of our continuous improvement efforts, the plan includes expanding the range of available models to cover a broader spectrum of user needs.

6. Machine Passport Use

6.1 Overview

The Machine Passport serves as an integral component within the industrial data ecosystem, meticulously designed to curate and administer the extensive data generated in the use phase of the industrial machinery. Also, the Machine Passport can be seen as an authoritative digital repository that meticulously archives all pertinent information related to the machinery, ensuring comprehensive accessibility.

For professionals in the field, the Machine Passport simplifies the complexity of data management. It efficiently aggregates dynamic data, such as sensor outputs in real-time operations, alongside static data, encompassing design schematics, production records, and operational logs. This system acts as an intelligent intermediary, negating the necessity for stakeholders to navigate the complexities of data storage systems.

The Machine Passport's utility spans across various operational roles. Engineers can readily access updated design details, production managers can oversee manufacturing processes with enhanced oversight, and sustainability officers can efficiently track recycling initiatives. Facilitated by a RESTful API interface, the Machine Passport ensures prompt and precise data retrieval.

At its core, the Machine Passport employs sophisticated Explainable AI (XAI) algorithms within its array of solutions to enrich the interpretability and applicability of data-driven insights. XAI demystifies the decision-making processes of AI, providing stakeholders with transparent and understandable data interpretations. This transparency is crucial for refining machinery design, preventive scheduling maintenance, and optimizing end-of-life strategies.

By converging multifaceted data streams and harnessing the power of intelligent analytics, the Machine Passport assures that the wealth of accumulated data translates into actionable intelligence. This, in turn, fosters innovation and operational excellence in the realm of industrial activity.

6.2 Features

The Machine Passport platform is endowed with a suite of advanced features and functionalities tailored to enhance user engagement and data interaction. Here's a detailed enumeration of its capabilities:

- User Interface (UI): A user-centric, intuitive interface that streamlines interaction and navigation. The UI has been crafted to simplify complex data into user-friendly formats, enabling users from various technical backgrounds to operate with ease.
- Visual Analytics: Leveraging cutting-edge visual analytics, the platform presents data through an array of dynamic charts and tables. This visualization capability transforms raw data into visual stories, making the analysis more comprehensible and actionable.
- Database Connectivity: With the flexibility to connect with diverse database architectures, including both SQL and NoSQL databases, the platform ensures seamless integration with

existing data storage solutions. This adaptability allows for a broad spectrum of data to be accessed and utilized, irrespective of the database format.

- Excel File Integration: The system provides the capability to import data directly from Excel files. This feature is particularly beneficial for users who manage datasets in spreadsheet formats, allowing for a smooth transition and amalgamation of data into the platform.
- Custom Dashboard Creation: Users have the autonomy to craft their dashboards, tailored to their specific requirements. This personalization feature empowers users to curate the data presentation that resonates most with their role and objectives, enhancing their decision-making process.
- Data Management and Sharing: At the heart of the Machine Passport is its robust data management system. It not only securely stores and organizes data but also verifies, validates, and maintains a historical record of formats across various data schemas from each solution thanks to the "Data Harmonizer". It also facilitates controlled sharing, ensuring that the right information is accessible to the right stakeholders at the right time.
- Explainable AI (XAI): The platform incorporates XAI algorithms to provide clarity and understanding behind AI-driven decisions. This feature is instrumental in fostering trust and transparency, enabling users to grasp the rationale behind predictive analytics and machine learning outputs.
- Multi-Source Data Acquisition: Recognizing the multi-dimensional nature of industrial data, the Machine Passport is designed to aggregate and process data from multiple sources, offering a comprehensive view of the equipment's lifecycle.

6.3 Technical specifications

The Machine Passport platform is a comprehensive server-side solution built on the robust foundation of ExpressJS, a web application framework for Node.js. This choice ensures high performance and flexibility, making it a suitable back-end for systems demanding efficient data handling and real-time responses.

For rendering the user interface, the system utilizes Handlebars (hbs) templates, a powerful templating engine that allows for the creation of semantic templates effectively. This templating approach promotes a clean separation of the server logic from the client-side interface, facilitating easier maintenance and scalability.

The visualization layer of the Machine Passport employs PlotlyJS for charting functionalities. PlotlyJS is a high-level, declarative charting library that provides rich, interactive charts. It is especially adept at turning complex datasets into compelling and informative graphics, enhancing the user's data interpretation capabilities.

For tabular data presentations, the platform integrates DataTables, a plug-in for the jQuery Javascript library. DataTables is a highly flexible tool that seamlessly creates interactive tables, which can integrate with data from various sources, enhancing the user experience with options like instant search and page navigation.

The connectivity with SQL databases is managed through Sequelize, a promise-based Node.js ORM (Object-Relational Mapping) for Postgres, MySQL, MariaDB, SQLite, and Microsoft SQL

Server. Its solid transaction support, relations, eager and lazy loading, read replication, and more, provide a full-featured suite for managing SQL database schemas and data.

In the realm of NoSQL database interactions, Mongoose is employed. Mongoose is an elegant MongoDB object modeling for Node.js, offering a schema-based solution for modeling application data. It includes built-in type casting, validation, query building, and business logic hooks, which makes the handling of MongoDB operations more structured and integrated into the application workflow.

For the importation and parsing of Excel files, the platform utilizes the XLSX reader. This utility is instrumental in interpreting and ingesting data from .xlsx files, enabling users to easily upload and integrate spreadsheet data into the Machine Passport's database for further processing and visualization.

For data validation and verification, the "Data Harmonizer" module has been developed. This module acts as a proxy for Machine Passport communications, allowing the data received to be verified with the data schemes corresponding to each solution. It allows storing both the data received and the different schemas or files within a Minio and a MongoDB.

The combination of these technologies is strategically chosen to not only ensure a high degree of reliability and performance but also to foster a modular and scalable environment. This technical composition empowers the Machine Passport platform to adeptly handle complex data processes, rendering it an indispensable tool in the realm of industrial data management and analysis.

6.4 Implementation status

6.4.1 Current implementation

The Machine Passport platform is progressing steadily towards its full functionality, with several key milestones already achieved. The current status of implementation includes the following advancements:

- Library Integration: All necessary libraries essential for the development of the platform have been successfully imported. This foundational step ensures that the development team has access to a suite of tools and functions that will drive the creation of robust features within the platform.
- Server Establishment: The platform has been established as a server capable of handling API communications, which forms the backbone of the Machine Passport's operational capabilities. This allows for seamless interaction between the frontend and backend, facilitating data exchange and processing.
- User Interface Development: A basic user interface has been developed, providing an initial layout and structure for users to interact with the platform. While this is an early version, it lays the groundwork for a more sophisticated UI that will be built upon with additional features and functionalities.
- Excel Integration: Functionality to import and process Excel files has been implemented. This allows users to feed data into the Machine Passport directly from Excel spreadsheets, significantly easing the data integration process from various sources.

- MongoDB Connectivity: The platform can now connect with a MongoDB database, enabling the storage and retrieval of data. This connection is vital for managing the large and complex datasets typically associated with manufacturing processes.
- Correlation Algorithms: Algorithms to create correlations from data imported from Excel files have been developed. These algorithms are crucial for identifying relationships between different data points, which can be visualized in the charts, providing insightful analytics to end-users.
- Data Tables: The creation of dynamic tables using data from Excel files is now possible. This feature allows users to organize, view, and analyze data in a tabular format, offering a clear and structured representation of data for better decision-making.

The current phase of the Machine Passport's development is focused on establishing a robust and flexible infrastructure, ensuring the system's core functionalities are in place. As the project advances, further enhancements and refinements will be made to these components, moving towards a comprehensive solution that fully embodies the envisioned capabilities of the Machine Passport platform.

The current version of the AIDEAS Machine Passport solution can be found in https://gitlab-cigip.alc.upv.es/aideas/machine-passport/mp.

6.4.2 Next developments

As we look ahead, the Machine Passport platform is gearing up for a series of strategic advancements that will significantly enhance its capabilities and user experience. The future development plans include:

- Integration with MongoDB: The upcoming phase will focus on the creation of charts and tables directly from data residing in MongoDB databases. This will enable real-time visualization and analysis, providing users with powerful tools to interpret complex data sets.
- SQL Database Connectivity: Establishing connections with SQL databases is on the horizon. This expansion will allow the platform to access and visualize data from a wider array of database systems, ensuring versatility and broader applicability across different manufacturing environments.
- Chart Categories Expansion: We will enhance the range of available chart categories, offering users a diverse suite of visualization options. By doing so, users can choose the most appropriate and insightful ways to display their data, tailored to specific analytical requirements.
- Security Enhancements: Security measures will be a focal point of future development, ensuring that data integrity and access control are maintained at the highest standards. These measures are crucial for safeguarding sensitive manufacturing data and maintaining user trust.
- Backend Solution Integration: Efforts will be made to establish seamless communication with backend solutions. This integration is pivotal for a synchronized and efficient workflow, enabling the backend algorithms to process data and feed actionable insights back to the frontend.

- Knowledge Visualization Functionalities: The development of functionalities to visualize the knowledge produced by backend solutions is planned. These functionalities will translate complex data analyses into understandable and actionable information, directly within the user interface.

By focusing on these key areas, Machine Passport is poised to become a more robust, secure, and user-friendly platform. The enhancements will drive smarter data integration, informed decision-making, and optimized performance, aligning with the goal of supporting the entire lifecycle of industrial equipment concerning the use phase of the machine.

7. Conclusion

This deliverable presents a technical summary of the solutions included in Work Package 5 of the AIDEAS project. It corresponds to the first release of the solutions, in M18, there will be two additional iterations of this deliverable, one in M24 for the final release of the solutions and a last one at the end of the project including possible changes derived from the continuous integration and the validation of the solutions inside the Pilots. As mentioned in the executive summary, further complementary information on the solutions can be found in D2.2.

The document describes the solutions in this Suite, designed to address different needs in the lifecycle of industrial machinery, namely:

Machine Calibrator: The AI^{MC} optimises machine performance by adjusting its operating parameters.

Condition Evaluator: The AI^{CE} continuously evaluates the condition of machines to prevent breakdowns and improve operational efficiency.

Anomaly Detector: The AI^{AD} is an advanced toolkit for detecting component or machine-level anomalies during factory operations.

Adaptive Controller: The AI^{AC} uses advanced machine learning algorithms to optimise machine operation, minimise human error and increase efficiency by automating tasks.

Quality assurance: The AI^{QA} solution focuses on detecting and locating visual defects in products using computer vision techniques.

It also includes information related to the **Machine Passport** solution from a use point of view, given that it acts as a comprehensive digital repository for industrial data, simplifying data management and providing real-time access to dynamic and static information about machines.

Altogether, the Use Suite solutions represent significant advances in automation, fault detection and data management in industrial environments, contributing to improved efficiency, quality and safety in industrial production.

The development of the solutions goes according to the plan with a first stable version ready to be tested. These initial versions are available through the Gitlab repository as indicated in the Current implementation sections above.

In addition to all the next developments included in detail along the document, it is worth noting that demos for each of the solutions will be prepared, so that it is clearly and easily seen the way they work and the problems they address. These demos will serve as a way to show what the solutions offer in different dissemination activities, but also to gather feedback from the potential users (including project's pilots) and also from the members of the Advisory Board. In this way the solutions will be tested and validated from this early stage. In fact, the solutions in this Suite will be implemented in the project pilots in different sectors: machining centres (PAMA), blow moulding machines (BBM), stone cutting (D2Tech) and visual inspection machines in food sector (MULTISCAN).