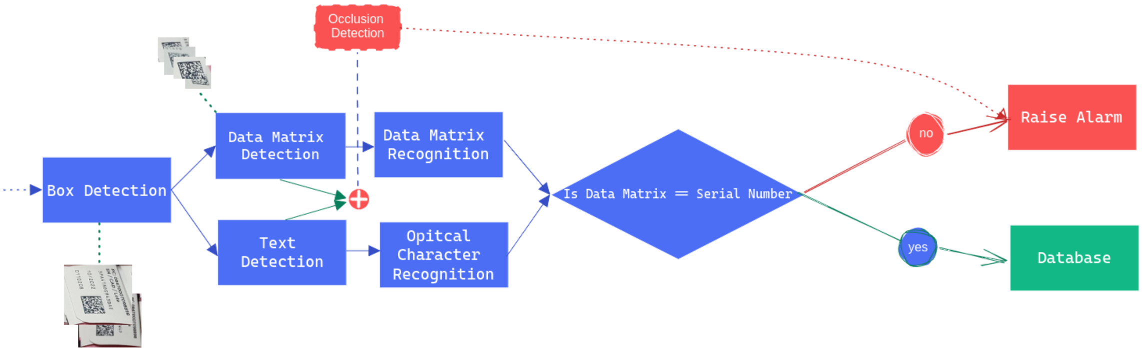
Technical Specification Double-side Page

1. **TECHNICAL SCOPE:** Summarize the mock-up devised during the EXPLORE phase: how have you addressed the challenge/Theme Challenges and tackled with its requirements and data. Include a diagram.

During the explore phase, we devised a mock-up solution to address the challenge of extracting printed text information and data matrix code from a series of boxes and matching the serial code in the data matrix with the printed text information. Our solution involves a combination of text detection, optical character recognition (OCR), data matrix detection, data matrix recognition, and box detection. Since it is necessary to implement the final solution on a CPU-based system or an edge device, we based our pipeline parts on light solutions.

First, we detect the boxes in the input image using our box detection algorithm. Then, we extract the printed text information using our text detection and OCR algorithms. Next, we detect the data matrix codes using our data matrix detection algorithm. Then, we recognize the serial code in the data matrix using our data matrix recognition algorithm. Finally, if the printed serial numbers are not equal to the data matrix, we send an alarm to the supervisor. Moreover, we send the location of the boxes that have any kind of occlusion (partial text or data matrix). The diagram below illustrates the overall process flow of our solution:



2. **ALGORITHMS, TOOLS AND CONCLUSIONS:** Detail the algorithms and tools identified to accomplish the challenge/Theme Challenges. Show clear understanding of the used REACH dataset/s and addressed challenge/Theme Challenges.

To accomplish the challenge of extracting printed text information and data matrix code from a series of boxes, and matching the serial code in the data matrix with the printed text information, a combination of algorithms and tools were employed. These methods were primarily based on state-of-the-art deep learning techniques and leveraged the REACH dataset for training and evaluation purposes.

- Box Detection: Initially, we had to extract the bounding boxes from the provided images, as the polygon annotations with the image size. Custom algorithms were developed to accurately identify and extract the boxes in the image. Text Detection: To locate and extract the printed text information from the boxes, a deep learning-based text detection algorithm was employed. This algorithm was trained on a relevant dataset to recognize and locate printed text regions within the image.
- Optical Character Recognition (OCR): Once the text regions were identified, an OCR algorithm was utilized to convert the detected text regions into machine-readable text. OCR models are typically trained on large datasets containing various text examples, ensuring high accuracy in recognizing characters and text patterns. Our method is deploying Transformers, ResNet, and Attention mechanisms to enhance accuracy.
- Data Matrix Detection: A data matrix detection model was trained based on available datasets with data augmentation. This model was designed to identify and extract data matrices in the provided image, allowing for further analysis and processing.
- Data Matrix Recognition: After detecting the data matrices, a data matrix recognition algorithm was used to decode the information contained within the data matrices, including the serial codes.
- Matching Process: The final step involved comparing and matching the extracted serial codes from the data matrices with the corresponding printed text information. This was achieved through the implementation of custom matching algorithms, ensuring a high degree of accuracy in identifying the correct pairs.

These deep learning-based solutions were designed to operate efficiently on CPU-based systems, making the overall process accessible to a wider range of hardware configurations. By employing a combination of text detection, OCR, data matrix detection, and data matrix recognition algorithms, we were able to successfully address the challenge of extracting and matching printed text information and data matrix codes from a series of boxes. The utilization of the REACH dataset, along with additional datasets, played a crucial role in training and evaluating the performance of these algorithms and tools, resulting in a comprehensive and accurate solution to the challenge.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981

3. **SCALABILITY AND FLEXIBILITY OF THE SOLUTION:** Discuss whether the solution can truly cope with humongous and increasing datasets and how flexible it is to adapt to other related domains and integrate into Data Value Chains (DVC).

The solution presented is designed with scalability and flexibility, allowing it to cope with humongous and increasing datasets while also being adaptable to other related domains and easy to integrate into Data Value Chains (DVC).

1. Microservices Architecture: Implementing deep learning models as microservices ensures that each component can operate independently. This modular approach allows for individual services to be easily updated, replaced, or scaled without affecting the overall system. In the event that a specific model requires more processing time or resources, it can be duplicated, enabling the system to maintain high performance even with growing datasets.
2. ONNX Conversion: Converting the models to the Open Neural Network Exchange (ONNX) format provides a consistent and interoperable framework for deploying the models across various platforms and hardware configurations. This ensures that the solution remains flexible and can be easily adapted to other related domains or integrated into Data Value Chains.
3. Optimized Protocols: Utilizing optimized protocols such as gRPC, Redis, and message brokers like Kafka and RabbitMQ helps manage scalability by enabling efficient communication between microservices. These protocols also facilitate load balancing, fault tolerance, and overall system reliability, ensuring the solution can effectively handle large and increasing datasets.
4. Containerization: Implementing each microservice within a separate Docker container further enhances scalability, as containers can be independently scaled according to the resource requirements of each component. Containerization also ensures a consistent environment across different deployment scenarios, simplifying deployment and reducing potential issues.
5. Monitoring and Logging Tools: The incorporation of monitoring and logging tools such as Grafana and Loki supports the stability of the pipeline, allowing for continuous assessment of the system's performance and identification of potential bottlenecks. These tools aid in addressing issues before they escalate and impact the overall system's ability to process large datasets.

In summary, the solution's design focuses on scalability and flexibility by leveraging a microservices architecture, ONNX conversion, optimized protocols, containerization, and monitoring/logging tools. These features allow the system to efficiently handle large and increasing datasets, adapt to other related domains, and easily integrate into Data Value Chains.

4. **DATA GOVERNANCE AND LEGAL COMPLIANCE:** Describe the security level of the proposed solution, i.e. how authentication, authorization policies, encryption or other approaches are used to keep data secure. Explain how will be compliant with the current data legislations concerning security and privacy (e.g. GDPR).

We do not rely on personal data, and our solution complies with the General Data Protection Regulation (GDPR). We respect European policies regarding data protection and privacy, the developed system is password protected, and the data processing procedure is transparent. It should be noted that only the operator or authorized personnel can access the platform and the data is stored securely in a safe manner while utilizing encryption techniques. Moreover, we do not employ any personal or private data and we are completely GDPR compliant. In addition, only the defined users have access to the data (with limitations). All of the possible data leakages have been managed. No one else has access to the camera streams. Only the system administrator has access to databases and user management.

5. **QUALITY ASSURANCE AND RISK MANAGEMENT:** Describe the quality process planned for the final product. Technologically, which are the potential risks in all the phases of the project (design of the solution, development, testing, deployment...) and indicate mitigation plans to still fulfil the challenge/Theme Challenges and data provider requirements.

The first risk concerns the camera configuration and lightening of the captured images. Our extensive experiments demonstrated that our deep learning platform not only extracts printed text information and data matrix code from a photo taken by the provided samples, but also can work effectively with different types of the images. The second risk is the low AI computation time requested by the problem owner, while not allowing the utilization of very powerful computers at the same time. After thoroughly training our model, we will optimize it such that the computation time will be below one second for each photo on an ordinary CPU-based system. The third risk is the local implementation of the framework without any internet access. Our solution is developed such that it is not necessary to access an internet connection. The fourth risk is outliers and special samples not included in the training dataset. Through enlarging the dataset with similar data, we can improve the generalization of our trained model and overcome the limitations of the imbalanced parts of the dataset and unseen data. Biassed models and biassed training could be viewed as the fifth risk which could be mitigated through adversarial training. The sixth risk is due to changes in the imaging environment. This is an engineering problem that can be avoided by preparing different environmental conditions and feeding them as training data to the deep-learning platform. To manage registered risks, appropriate responses are provided as well as continuous risks' monitoring and evaluation processes. From the software development point of view, we have developed our backend and frontend by the standard DevOps solution with consideration of different tests such as unit test and integration test, etc. In addition, we completely developed our software solution in an advanced CI/CD pipeline which overcomes several risks in the software development procedure.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981