INTERNSHIP REPORT

Submitted to

Seminar für Sprachwissenschaft
Eberhard Karls Universität Tübingen

Internship at

Maddox AI GmbH

Tübingen, Germany

Name: Seoha Lee
Student ID: 6603078
Department: ISCL
Supervisor (University): Dr. Çağrı Çöltekin
Supervisor (Company): Alexander Böttcher

Internship Period: Apr 2025 – Jul 2025

Date of Submission: October 11, 2025

Contents

1 Introduction

From March to July 2025, I completed a four-month internship at Maddox AI GmbH. This internship was a continuation of my previous role as a working student at the company. While my earlier responsibilities centered on managing training data for vision-based AI models, the internship allowed me to participate in an ongoing project focused on developing an AI model for defect detection using physical markers. This approach serves as an alternative to direct visual defect detection, enabling simpler and easier data annotation for non-AI experts.

This report begins with an introduction to the company and an overview of my responsibilities as a working student. Then, it provides a detailed explanation of the physical marker detection project, including the development procedure of the physical marker detection model. Finally, I reflect on the key insights gained through the project and conclude with my overall internship experience.

2 Company Overview

Maddox AI is a vision-based quality control startup specializing in artificial intelligence solutions. The company develops software products for AI-driven defect detection and offers customized hardware solutions, including tailored camera systems that enable customers to capture optimized image data for AI model training. I joined Maddox AI as a Data Annotation Specialist from March 2024 to August 2025, gaining hands-on experience in AI software and data management workflows.

2.1 About the Company

Maddox AI has three offices, including the Tübingen office where I worked, as well as locations in Cologne and Berlin. The company is organized into several teams, such as Customer Service, Machine Learning, Marketing, and HR. During my internship, I primarily collaborated with the Machine Learning (ML) team and the Customer Success (CS) managers. I worked closely with two ML engineers to discuss model performance and with the CS managers to incorporate customer needs into projects.

2.2 My Role as a Working Student – Data Annotation Specialist

Accurate data annotation is critical for training AI models. In a small company like Maddox AI, maintaining a sufficient volume of high-quality annotated images can be challenging. To address this, three working students are dedicated specifically to image annotation, ensuring a consistent and reliable dataset for AI training.

The core annotation work involves drawing polygons around areas where the AI model is expected to detect defects. My primary tasks included annotating images for the training set, correcting false alarms or missed detections after model training, and testing the model using sample images. Occasionally, I performed other tasks, such as recording sample products with the customized camera or translating help desk content between German and English. However, the majority of my work focused on data annotation.

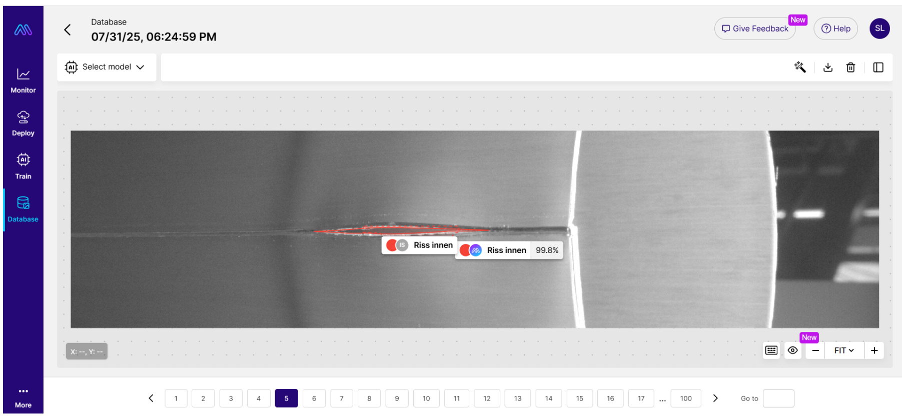

Figure 1: Data annotation using polygons on the Maddox AI platform

3 The Internship

While searching for an internship opportunity, I reached out to my supervisor, a Machine Learning Engineer at Maddox AI, to ask if there were any projects I could contribute to. Building on my previous experience as a working student in data annotation, we decided to focus on the physical marker detection project, which had been suggested by a third party.

3.1 Introduction: Physical Marker Detection Project

The third party approached Maddox AI with the idea of an alternative annotation method — physical marker detection. The purpose of physical marker detection is to make data annotation more accessible to a wider range of people, including those who may not be able to use a computer. For example, individuals with disabilities could participate in the annotation process by physically marking defects directly on products instead of drawing polygons on a screen.

In the traditional defect detection workflow, models are trained using polygons manually drawn around defect areas within the Maddox AI platform. In contrast, the physical marker detection approach involves two steps: first, training a model to detect the physical markers, and then using those detected markers to guide a second model that identifies the actual defects based on the marker's position.

Physical markers can take different forms, such as arrows, paper strips, or circles around defects. In this project, arrow markers were primarily used, where the arrowhead points directly to the defect. Towards the end of the project, we experimented with other marker types, such as attaching a paper strip and using a keypoint detection model to recognize its three corners, or simply drawing a circle around the defect.

3.2 Project Procedure

i Preparing Dummy Parts

The project began by generating toy defects on dummy parts to simulate real-world production defects in a controlled environment. The dummy parts were simple metal rings, and the defects were artificially created by hammering dents into their surfaces. Each side of a dummy part contained one or two synthetic defects. These artificially generated samples served as the foundation for the initial dataset used in subsequent stages of model development, including image recording, annotation, and training.

ii Image Recording and Uploading to the Maddox AI Platform

After creating the dummy defects, images were captured using an industrial camera together with the Baumer camera software. For each batch, approximately 300 images were recorded by slightly rotating the dummy parts under the camera, which ensured variation in angle and lighting conditions. These images were then uploaded to the Maddox AI platform.

To reduce manual effort, the recording and uploading process was partially automated. The image capture was handled through Baumer's `neoAPI` Python library, which enabled direct communication with the industrial camera. A custom script was written to connect to the camera, capture 300 images, resize them, and save them into a local folder. The script also logged the total recording time and calculated the average capture time per image, making the process more transparent and efficient.

For uploading, the `pymaddox` API (a Python client for the Maddox AI platform) was used. Authentication was automated by securely reading login credentials from local text files and passing them to the platform's authentication client. This allowed login and logout processes to be performed programmatically, without manually entering credentials each time. Once authenticated, image upload and download operations were handled directly via the API, removing the need to rely on the platform's graphical interface.

Together, these steps significantly streamlined the workflow: the combination of automated camera control and programmatic interaction with the Maddox AI platform reduced repetitive tasks, ensured faster data preparation, and improved reproducibility across experiments.

iii Physical(Arrow) Marker Annotation and Marker Model Training

Once the images were uploaded to the platform, the next step was to train a model to detect the physical markers—in this case, the arrows. To increase precision, the arrow markers were annotated by separating the head and tail into two distinct classes rather than treating the entire arrow as a single object. This approach made it possible to calculate the defect coordinates more accurately.

For the training process, Maddox AI's existing object detection framework was used. The model allows users to define multiple classes by drawing polygons around the relevant regions. In this case, two classes—"head" and "tail"—were created, and I annotated approximately 50 images focusing on clear and unambiguous arrow markers.

After training, the model's performance was evaluated using its confusion matrix. The results showed that the model could reliably detect most arrow markers. The main difficulties arose in cases where markers were partially obscured due to the angle of the dummy part or reflections from lighting. However, this limitation was not considered critical, since Maddox AI's clients typically rely on customized camera systems designed by professional engineers, which minimize such issues in real-world applications. For this reason, I chose not to annotate images with unclear markers, as doing so might have increased the model's sensitivity and led to unnecessary false alarms.

iv Loading the Physical Marker Detection Model and Preparing Image Transforms

The physical marker detection model was loaded using the ONNX Runtime framework. The trained model file was integrated into the inference pipeline by first creating a session options object and then initializing an inference session. This configuration allowed the model to be run efficiently during experiments.

Before passing images into the model, preprocessing steps were applied. For example, grayscale images were automatically converted into three-channel color images, ensuring compatibility with the model input format. Additionally, the images were normalized to maintain consistent input data quality. These transformations ensured robust handling of different types of input data during model inference.

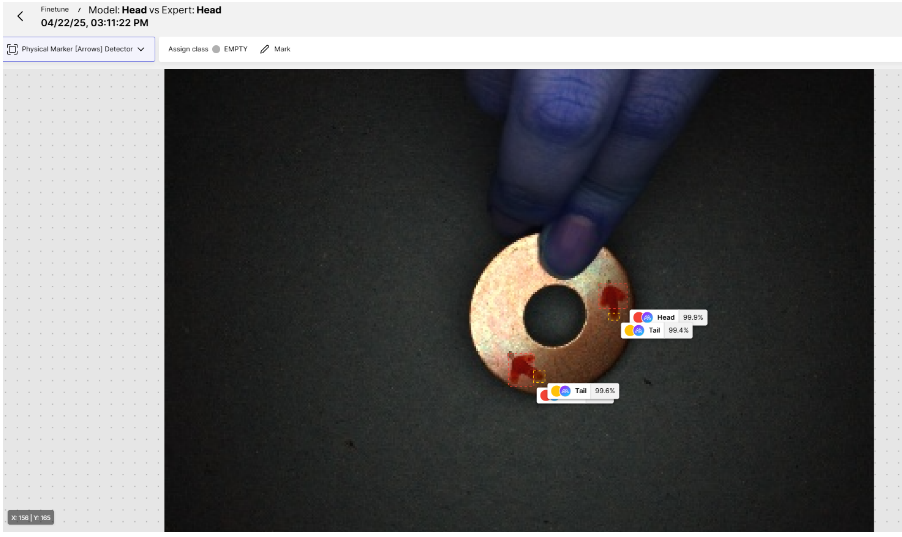

Figure 2: Detection of arrow markers after training

v Algorithm for Physical Marker Prediction

The model predictions included bounding boxes for different classes of physical markers. To associate head and tail markers, an algorithm was designed to compute the center point of each detected rectangle and then calculate distances between markers of different classes. For every head marker detected, the algorithm searched for the closest tail marker by comparing the Euclidean distance between their center coordinates. The pair of markers with the minimum distance was then selected as the corresponding head–tail pair.

This procedure ensured reliable pairing of physical markers and reduced the likelihood of incorrect associations between markers belonging to different arrows.

vi Downloading the Physical Marker Model Predictions

After setting up the model pipeline, predictions for images with physical markers were generated and downloaded. This process was automated using the pymaddox API, which not only handled image downloads but also retrieved metadata in JSON format. The metadata included polygon coordinates for both head and tail markers. These coordinates were subsequently used to calculate the area of defects, particularly focusing on the endpoint of the arrow head.

To validate the predictions, the areas corresponding to the head and tail markers were visualized. Visualization was carried out using matplotlib, allowing inspection of whether the predicted coordinates matched the intended marker locations.

vii Defect Localization Using Physical Markers

The first approach for defect detection was based on the coordinates of the physical markers. Specifically, the midpoint between the head and tail of each arrow was calculated, and a line connecting these midpoints served as a reference for identifying potential defect locations. This approach allowed the defect detection model to focus on relevant regions highlighted by the physical markers.

viii Defect Detection Model Training, Evaluation, and Marker Experiments

Since the physical marker project was a trial study, model evaluation after training was carried out primarily through visual inspection rather than strict quantitative metrics. The model performed reasonably well in predicting defects, though accuracy was sometimes limited by unclear marker visibility and occasional misdetections of irrelevant areas.

In the later stages of the internship, alternative marker designs were explored to improve localization accuracy. One experimental approach involved attaching paper strips with small arrows to products. The company's existing keypoint detection model was then used to identify three specific points on each strip. These three points formed a triangular configuration, enabling more precise defect localization compared to the linear arrow approach.

I recorded images of dummy parts with the new paper strip markers and annotated the three keypoint locations for training. However, due to the limited internship timeframe, I was unable to complete a systematic comparison between the original arrow-based detection and the keypoint-based method. Another proposed idea was to draw circular markers directly around defect areas, though this remained at the conceptual stage.

4 Current Stage of the Project

Following a meeting with the CTOs of Maddox and the third party, Maddox officially gained them as a customer. This marked a transition of the Physical Marker Project from a trial study to an industrial-level application. Building on the exploratory results of my internship, the company scaled up the approach by standardizing both the marker design and the evaluation process.

Three different sizes of 3D-printed physical markers are now used in combination with Maddox's customized defect camera device. These markers consist of pink strips with small squares at one end, where the defective area must be positioned within the square. This design simplifies annotation and reduces ambiguity, compared to the arrow-based approach that I worked on.

The workflow at this stage follows a refined version of my internship pipeline:

- **Marker detection:** Models are trained to reliably detect the 3D-printed square markers.

- **Defect localization:** The location of the small square provides a consistent reference point for defect annotation.

- **Defect detection:** Images of defects within the squares are annotated, and a dedicated defect detection model is trained and evaluated.

Unlike during my internship, where evaluation relied primarily on visual inspection, the current project employs standard quantitative metrics such as precision, recall, and mean Average Precision (mAP). This provides a more rigorous framework for assessing performance and ensures the reliability required for industrial deployment.

Although I was not directly involved in this commercial phase due to the conclusion of my internship, I recognize that the trial work I performed—especially in model loading, prediction, visualization, and exploration of marker types—helped to shape the company's decision to pursue standardized markers and formal evaluation methods for their partnership with the third party.

Figure 3: The 3D printed physical markers

5 Summary and Reflections

During my internship at Maddox AI, the Physical Marker Project provided my first hands-on experience with the full lifecycle of a machine learning application: from problem definition to data preparation, model training, prediction, and evaluation. This experience gave me insight not only into the technical workflows but also into the challenges of developing applied AI solutions for industrial use.

Key skills and insights I gained include:

- Practical experience with ONNX Runtime for loading and running detection models.
- Using Python APIs (such as pymaddox and Baumer's neoAPI) for automating image capture, uploading, downloading, and prediction tasks.
- Designing preprocessing pipelines, including grayscale-to-color transformations and normalization.
- Implementing algorithms for associating predictions (head–tail marker pairing).

- Visualizing and validating predictions using matplotlib.
- Experimenting with alternative marker designs (arrows, paper strips, and conceptual circular markers).

At the same time, the project highlighted areas for improvement:

- The lack of well-defined evaluation metrics limited the reliability of my conclusions.
- Time constraints prevented a full comparison between the original arrow-based method and alternative keypoint-based approaches.
- Limited exposure to customer requirements made it harder to align technical exploration with real-world needs.

Overall, the project was highly rewarding. It allowed me to bridge the theoretical machine learning knowledge I gained in the SNLP course with practical applications in computer vision and machine learning, while also highlighting the importance of structured evaluation and close integration with customer requirements. Although the project extended beyond the timeline of my internship, I hope that my contributions were helpful in supporting its progress.

References / Figure Credits

- Maddox AI Logo: from the Maddox AI platform
- Figures 1 and 2: screenshots from the Maddox AI platform
- Figures 3 and 4: photographs by Seoha Lee