Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 130 (2024) 387–397

www.elsevier.com/locate/procedia

57th CIRP Conference on Manufacturing Systems 2024 (CMS 2024)

A digital assistance system leveraging vision foundation models & 3D localization for reproducible defect segmentation in visual inspection

J. Koch*[,a], D. Jevremovic[a], K. Moenck[a], T. Schüppstuhl[a]

[a]*Hamburg University of Technology, Institute of Aircraft Production Technology, Denickestr. 17, 21073 Hamburg, Germany*

* Corresponding author. Tel.: +49-40-42878-4324. *E-mail address:* julian.koch@tuhh.de

Abstract

In response to the limitations of manual visual inspection in manufacturing industries, this paper presents a novel approach that combines Meta AI's Segment Anything Model (SAM) as a current representative of a vision foundation model with marker-based localization techniques to enhance quality assurance processes. Focusing on aircraft cabin components, where product complexity and variance pose significant challenges, a digital assistance system is developed. By integrating SAM-based segmentation and defect localization on the CAD model, the proposed system facilitates significant parts of the inspection workflow, thereby increasing efficiency and reproducibility. The system also enables the collection of comprehensive data sets, which is essential for the refinement of AI models, e.g. in transfer learning approaches. A two-pronged evaluation approach includes the assessment of localization accuracies in an experimental setup as well as the evaluation of usability and workload in an industry-oriented user study. The novel segmentation approach using the SAM achieved high usability, providing an efficient and user-friendly solution for the application domain, while maintaining a reasonable workload for the operators, highlighting its significant potential for streamlining industrial inspection processes.

Keywords: Digital Assistance System; Visual Inspection; Segment Anything Model; SAM; Defect Segmentation; Worker Assistance; Aircraft Manufacturing

1. Introduction

Quality assurance processes in safety-critical industries such as aerospace go beyond mere operational protocols; they are integral to certification, regulatory compliance, and ensuring safety standards. For aircraft in particular, where every component must meet the highest safety and quality standards, manual inspections are essential. However, these inspections present significant challenges: they are labor-intensive, require extensive documentation, and are subject to the variability inherent in human judgment. This is particularly problematic for aircraft interior inspections, where the quality of the visual inspection additionally affects customer perception and, therefore, competitive advantage [1].

Research to improve visual inspection has primarily followed two paths: automation and assistance. Automation attempts to replace the human element entirely with technology, using advanced image processing and Deep Learning (DL) to detect and classify defects. However, this strategy often falters in the face of the diverse, complex, and large-scale nature of aerospace components. Requirements for specific lighting conditions, physical accessibility to all parts of the product, and the need for extensive defect-specific training data make full automation impractical and economically unfeasible in many cases.

On the other hand, assistance-based approaches retain the human inspector as the central figure in the process, aiming to augment rather than replace their expertise [2]. Despite the potential of this strategy, current research falls short of providing effective tools for visual inspection, particularly in the field of surface damage segmentation, where accurate and reproducible defect identification is critical.

This paper presents a novel approach that significantly enhances the manual inspection process through a digital assistance system, emphasizing the value of human expertise while addressing the aforementioned shortcomings. We present the implementation of Meta AI's Segment Anything Model, a vision foundation model characterized by strong zero-shot capabilities. This allows for precise defect segmentation, enabling reproducible and user-independent defect masks. In tandem, we propose to combine this segmentation tool with a 3D localization method suitable for components with hard-to-reach inspection areas, placing defects in their exact spatial context.

10.1016/j.procir.2024.10.105

The remainder of this paper is organized as follows: First, we give an introduction to the activities and potential defect classes in the visual inspection of aircraft interiors and discuss the related work of assistance within these processes in section 2. Subsequently, we present the concepts (section 3) of the localization and segmentation tools of our assistance system and explain the corresponding implementation (section 4) as a mobile web application. This will be used to characterize the features in two separate evaluation processes. Once in terms of achievable defect localization accuracies, and once to evaluate usability and workload (section 5). After a discussion of the evaluation (section 6), the paper concludes with a summary and outlook (section 7).

2. Essentials & Related Work

2.1. Visual inspection

Visual inspection is a subjective inspection method in which the actual condition of a device under test (DUT) is observed by the human eye and compared to the expected target condition. Due to its flexibility and ease of implementation, it is frequently applied in aircraft manufacturing [3] as well as in their maintenance, repair and overhaul (MRO) [4]. Keferstein et al. [5] divide the activities performed as part of a visual inspection into four groups, which are represented in the upper part of Figure 1.

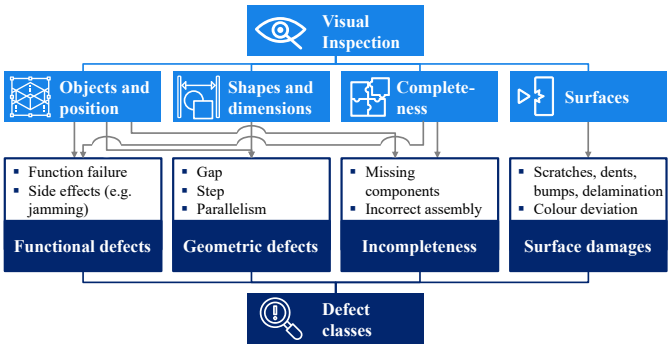


Fig. 1: Visual inspection activities (light blue) and associated defect classes (dark blue).

Unlike the basic activities, there is no uniform definition of defect classes since they are highly dependent on company and product specific conditions. For this reason, defect classes for this paper are defined on the basis of other work dealing with the visual inspection of aircraft cabin components. The underlying work [1, 6, 7, 8, 9] allows the derivation of four defect classes. With respect to the activities mentioned above, a *1 to n* relationship between the activities and the defect classes can be seen, which is represented in Figure 1 by directed edges. In addition, several examples of defect classes are given, which can often be found in the literature.

Due to its manual nature, the results of a visual inspection are highly dependent on the person performing the inspection and are therefore not error-free. Individual personal aspects such as care, experience, attention, as well as environmental conditions such as lighting and contamination can be potential sources of error [10]. Not only the process itself, but also its documentation contains sources of error, namely the widespread practice of handwritten documentation of inspection results in the aircraft industry [11]. Identified defects are often noted in an unstructured form on technical drawings or separate paper documents [12].

In addition, documentation is often converted to a digital format at a later date. However, this media discontinuity is also prone to errors and requires considerable manual effort [13]. Therefore, Digital Assistance Systems (DASs) are increasingly being used for these processes, which enable digital data acquisition and additionally reduce variance through targeted user guidance. These are described in the following section.

2.2. Digital Assistance systems for visual inspection

DASs for the visual inspection can be categorized according to their functional scope with regard to the activities shown in Figure 1. A distinction is made between systems with a more holistic approach and those designed to support a specific activity. In this section we focus on the domain aircraft industry.

For the completeness check of aircraft, Ben Abdallah et al. [14] present a a camera-based system that can be guided by either a robot or a human. The robot is suitable for inspecting small areas, while the tablet can be moved by a human to parts of the component which are hard to access. The manual version of the system includes a tablet as a human-machine interface and an additionally mounted industrial HD camera. In both cases, the completeness check is performed by first loading the component's CAD model and performing an edge analysis. The edges of the CAD model are then compared to edges extracted from the recorded videos. The tablet's pose detection enables the correct comparison of the relevant edges. As an output the system generates an inspection report with parts which are missing or incorrectly positioned. It should be emphasized that the human is involved in this process only as a mobile unit and not because of his expertise.

Building on this work, Hu et al. [9] present an approach that also starts from a CAD model and incorporates both localization and guidance features. Instead of using edge-based detection, it employs DL algorithms to detect missing objects. The use case evolves around the completeness check of cable clips during aircraft assembly. With the help of the implemented DL approach, the performance of object detection in challenging assembly scenarios, such as bright light conditions, proves to be more robust. Instead of a tablet, a smartphone is used, which is also manually guided.

In addition to the approaches in which the human serves only as a movement unit for the camera, Marino et al. [15] present an augmented reality solution for visual inspection based on the same technology components (CAD model, tracking, localization). The CAD model is superimposed on the real component on a tablet. This allows the inspector to detect deviations from the design data directly and to annotate them by creating points on the object with ray casting functions. The user's expertise

is actively engaged to provide additional information. This application is not limited to the completeness check as different defect descriptions can be stored in the annotation.

In summary, the work presented is mainly concerned with the overlay of a 3D model, which is essentially used for automatic or manual detection of primarily missing and incorrectly assembled components. There is a lack of work in the literature that focuses on the assistance of manual inspection of surface damage. Therefore, the goal of this work is to fill this gap and to develop a system that can be used for surface damages. At the same time, it will be shown that the full range of functions (e.g. functional checks) of a visual inspection can be integrated into the system. To this end, we are building on the results of our first system iteration, which already provided a simple way to annotate different defect types [16].

3. Digital assistance system for visual inspections

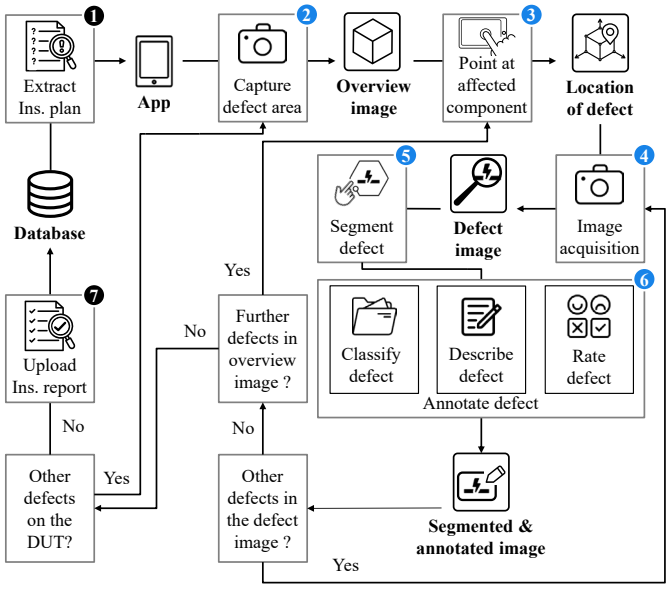
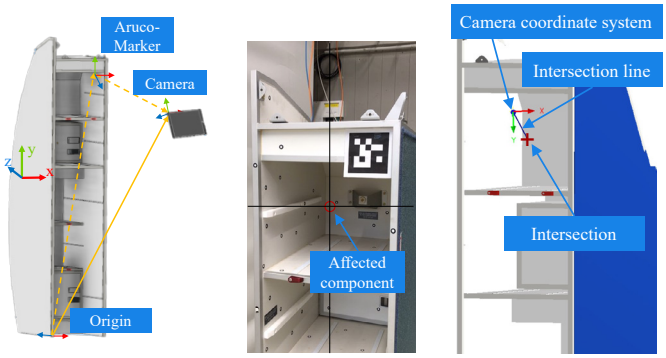


Fig. 2: Flow chart of the digital assistance system for visual inspection. Process steps for defect localization, image acquisition, segmentation and annotation marked in light blue.

As in our previous system iteration, we are designing the DAS as a mobile web application that runs on a tablet. This selection is in line with study results from Piontek et al. [17], who identified tablets as a particularly accepted and effective technology for DASs in aircraft production. Following the procedure model of automated visual inspection of Keferstein et al. [5] we divide the process of our assisted visual inspection (Figure 2) into the essential sub-steps of defect localization (2+3), defect image acquisition (4), segmentation (5) and annotation (6) (classification and detailed description). The additional steps are necessary to integrate our system into higher-level information systems, allowing the extraction of initial inspection plans (1) and the return of the final inspection report (7). In the following, we explain the process flow along the process chain with a focus on the conceptual design of the localization and segmentation solution modules.

3.1. Marker-based 3D defect localization



(a) Determining the camera pose. (b) Point input on the acquired image. (c) Determination of the component.

Fig. 3: Illustration of 3D defect localization using an aircraft cabin component.

After initiating the visual inspection to be performed, the process begins with the inspector's evaluation of the DUT. If a defect is found, the first step is to document its location on the DUT (steps 2, 3). This is accomplished by combining marker-based localization with intersection point calculation. A reference to the CAD model is created by placing one or more markers on or around the DUT at known locations. The marker is captured by the tablet's camera and the tablet's position and orientation relative to the DUT is determined (Figure 3a). With a marker in the camera's field of view, the inspector takes a photo of the defect area, which is referred to as "overview image" in Figure 2. This image is displayed on the user interface and the inspector selects the affected component with his finger. This input is used to generate the pixel coordinates of the defective component on the 2D image (exemplary shown in Figure 3b). Given the camera pose relative to the model and the pixel coordinates in the 2D image plane, the inverted projection of the pinhole camera model can be leveraged to derive the corresponding 3D point in the CAD model. Thereby, a straight line can be defined which starts at the camera position and extends in the direction of the camera orientation. The first intersection of the line with the model can be determined and represents the link between the defect in the image and the 3D model (Figure 3c).

The described approach diverges from current methods by separating defect localization from direct defect image acquisition. By marking defect locations in an overview image taken from a distance, it ensures accurate assignment even in inaccessible areas, eliminating reliance on other localization methods prone to error accumulation, like using tablet IMUs. Additionally, it circumvents shortcomings of methods such as edge detection when capturing detailed defect images close to the component surface.

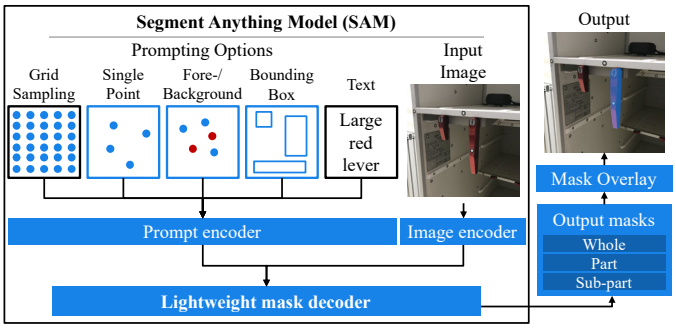


Fig. 4: Structure and prompting methods of the SAM and superimposition of a mask over an example image.

3.2. Zero-shot segmentation with Vision Foundation Models

After defect localization, inspectors move closer for detailed defect image capture, crucial for segmentation. The system requires an intuitive, efficient marking method for quick and accurate defect identification. It aims for consistent results across users, reducing subjective marking differences and enhancing data quality. This is particularly important for surface damages, where defect contours provide valuable training data for automated detection methods (further discussed in section 7). Classical image processing segmentation methods are not viable due to tablet hand guidance and expected defect contour fluctuations. Traditional DL methods rely on large, domain-specific datasets typically lacking in manufacturing [18], like the aircraft industry. As a result, more generalistic approaches are gaining traction, notably Vision Foundation Models (VFMs). VFMs leverage pre-training on extensive datasets to generalize across tasks, including object segmentation and classification, without domain-specific training. These approaches include zero-shot, requiring no target domain data, and few-shot, needing only a small dataset from the target domain.

One prominent work is the Segment Anything project by Meta AI [19]. It stands out for its extensive dataset, comprising over one billion segmentation masks on eleven million licensed images, along with the Segment Anything Model (SAM). The SAM, leveraging a Vision Transformer architecture, acts as a masked autoencoder capable of generating three hierarchical, pixel-accurate masks (whole component, part, sub-part) from input images. The masks are generated using a number of different prompting methods that can be superimposed on the input as shown in Figure 4. In a pre-study, zero-shot capabilities for segmenting domain-specific images across various aircraft production and MRO scenarios were evaluated. SAM exhibited high precision in generating output masks for both component contours and a diverse range of defects, including cracks, dents, and contamination [20]. Based on these capabilities, the SAM as a segmentation tool enables the support of different activities in the visual inspection and is therefore selected for the assistance system.

Additional tools are used for the further annotation of the segmented defect (step 6 in Figure 2), such as the assignment to a main and sub-defect class (according to Figure 1), the possibility of a free text description and the evaluation of the severity of the defect, which is relevant for subsequent processes such as reworking or scrapping. These were adopted from our previous work and are described in more detail in [16].

4. Implementation

This section focuses on the implementation of the localization and segmentation features within the DAS. The overall architecture follows a client-server model, which means that more computationally intensive applications can be easily outsourced and the overall application becomes more independent of the tablet currently in use. The front end of the application is designed as a web app, utilizing using React JS [21] as a library in conjunction with the MUI React Component library [22]. It should be noted at this point that the seamless use of the application places requirements on the server hardware used. For this work, a high-performance desktop computer equipped with an NVIDIA GeForce RTX 4080 graphics card, 64 GB RAM and an Intel Core i9-13900K processor was deployed. In particular, the calculation of the image embedding of the SAM requires a powerful graphics card, while the provision of multiple endpoints for communication between the front and back end entails a high RAM requirement. If the hardware is too weak, there may be delays in calculating the image embedding, which is a prerequisite for the segmentation process. This may have a negative impact on usability.

4.1. Part prediction with AruCo markers

Two web services are developed on the server for defect localization. The aim of the first web service is to check whether an ArUco marker (AM) [23] is recognized in the tablet camera's field of view. The video stream from the tablet camera is used as input via a P2P connection. OpenCV is used to recognize the marker in the frames. As marker recognition is a prerequisite for localization, the recording of the overview image (step 2 in Figure 2) is blocked on the front end until a marker is found. If an overview image is taken and a component is marked, this information is forwarded to the second web service, the aim of which is to identify the component concerned. To do this, the calibration data of the camera and the pose of the AM in relation to the component must also be known. The camera is calculated relative to the AM using OpenCV. This information can then be used to create the intersection line based on the pinhole camera model. To determine the intersection point, the straight line and the CAD model are imported into OpenCascade [24] and a collision check is carried out between them. The collision results show the intersection points of the straight line with the model surface. By measuring the distances between these intersection points and the camera position, the specific surface of the component that contains the intersection point with the smallest distance can be identified.

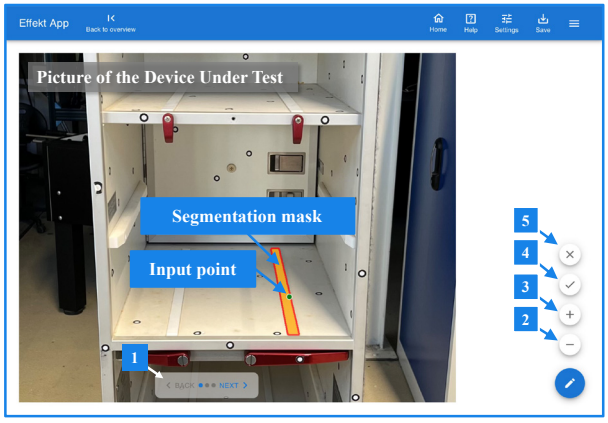


Fig. 5: GUI on the tablet after a single point prompt with the superimposed segmentation mask. Tools for switching between the three segmentation mask (1), deleting (2) and adding (3) of points, accepting (4) and discarding (5).

4.2. SAM-supported defect segmentation

As with localization, a web service is developed for segmentation for which the SAM was adapted accordingly. After acquiring the defect photo different prompting methods (marked in blue in Figure 4) can be used on the front end to segment the input image. For the first implementation, we assume that point-based prompting (single, foreground/background) is particularly intuitive, as the user indicates the defect with their finger. In addition, bounding box prompting is implemented, as this is widely used in computer vision, for example for drawing regions of interest. Figure 5 shows an example the user interface after single point prompting of a rail on a cabin component. The user has the option of selecting a suitable mask from the masks generated by SAM (1) or adding (2) or removing (3) areas via additional points (corresponds to foreground/background prompts). All user inputs are sent to the server with the corresponding command and pixel coordinates, from which the corresponding masks as a combination of vectors is created and transferred back to the front end for superimposition. If the segmentation mask is of sufficient quality, it can be accepted, which opens a window for further annotation of the error. If an incorrect component or defect is marked, the marking can be discarded (5).

Within the annotation window, the defect class can be selected from a drop-down list that includes the four classes shown in Figure 1. This ensures that the application can be used not only for surface defects, but also for other classes. However, a limitation of using the SAM for this application is that missing components simply cannot be marked due to the functional principle. In this case, alternative manual segmentation tools from the first system generation can be used. Nevertheless, these are very slow due to the large number of user interactions required. Therefore, there is potential for improvement in the completeness checking of defect classes, which is discussed in more detail in the section 7.

5. Evaluation of the assisted visual inspection application

This section deals with the evaluation of the developed application and its specific functions. Particular attention is paid to the two main functions: defect localization and defect segmentation. For this two separate evaluation processes are designed and conducted.

5.1. Evaluation of the localization accuracy

5.1.1. Experimental setup

This evaluation aims to assess the accuracy of tablet localization and interface calculations for defect localization in the 3D model. The experimental setup introduces a reference system comprising an industrial robot arm mounted on a mobile platform, specifically the UR10e from Universal Robots [25], and a motion capture system, the OptiTrack (OT) system from NaturalPoint Corporation [26].

The OT system utilizes markers to detect the position and orientation of objects in a room, while the robot arm guides the tablet to predefined spatial waypoints. The experimental setup is completed by an test object on which both an AM (90x90mm) and a target circle are attached. The robot platform and the test object are placed in the coverage area of the OT as shown in Figure 6a and referenced to each other by attaching markers to known defined points on the platform and the around the AM on the test object. This results in the coordinate systems of the AM (ACS) and the robot platform (PCS) in the global (OT) coordinate system. From hereon, static transformations can be made from the PCS to the robot coordinate system (RCS) and the camera coordinate system (CCS).

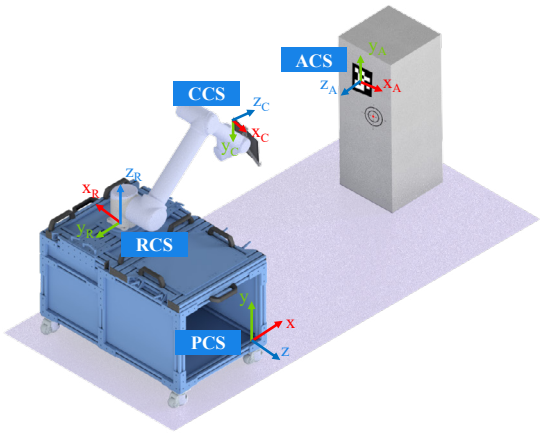
5.1.2. Experiment execution

The robot was moved to 15 points to carry out the test. Three planes were each defined in a cross shape (5 points per plane), with the center of the cross aligned as centrally as possible to the origin of the ACS (see blue points in Figure 6b). To ensure that the AM is always completely in the field of view of the tablet camera, the corresponding robot joint is additionally adjusted at each approached point so that the camera is aligned with the AM. At each point the pose of the camera is saved and a photo of with the tablet camera at highest resolution (4032x3024 pixel) is taken.

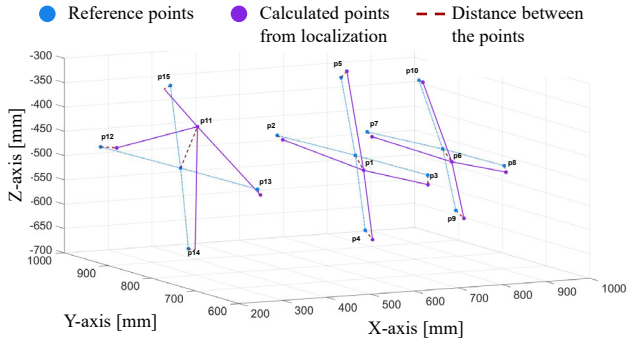
5.1.3. Results

For evaluation, the poses at the defined points of the ground truth system and the AM localization solution were transferred using transformations into the PCS, which represents the common coordinate system. To evaluate the accuracy of the localization and intersection calucaltion, the indicators of position accuracy (trueness) and position repeatability (precision) are used in accordance with the ISO 9283 [27]. A detailed description is given in Appendix A.

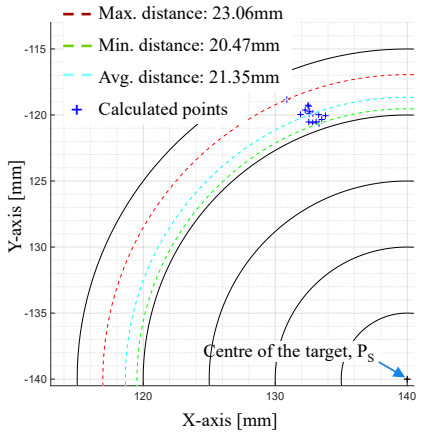
First, the position accuracy of the localization method is calculated. For this purpose, the Euclidean distance between the coordinates determined by the AM localization and the reference point is measured (see Figure 6b). Additionally to that the



(a) Experimental setup with the coordinate systems of the different systems.



(b) Illustration of the calculated camera positions in comparison to the reference points (in the PCS). The red dashed line represents the distance between the individual points.



(c) Distribution of the individual intersection points on the target (shown in the ACS). Point P_S describes the center of the target. The intersection points can be recognized as blue crosses in the graph.

Fig. 6: Setup and visualization of the results of the evaluation of the localization accuracy.

mean orientation error is calculated. To compare orientation of the reference and the localization system, a rotation matrix is determined that represents the rotation between the two orientations. By converting this rotation matrix into a rotation vector and angle, a meaningful comparison value can be determined. The evaluation results in an average deviation of the position of **26.37 mm** and the orientation of **2.61**°.

Indicator	Value
Trueness	21.34 mm
Precision	1.10 mm

Table 1: Accuracy indicators for the intersection calculation.

Secondly, the intersection calculation is investigated. To identify the affected component, the intersection point between the straight line from the camera origin, which is defined by the selected pixel in the image, and the 3D CAD model is determined (see section 4). For each point approached, the corresponding pixel coordinates of the target center point and the calculated camera pose are imported into the intersection calculation. The straight line is created on the basis of the data from the camera localization generated by the detection of the AM. The calculated intersection points for each approached point relative to the target position of the target P_S are visualized in Figure 6c. To evaluate the accuracy, the correctness and precision of the intersection calculation is determined resulting in values shown in Table 1.

5.2. Evaluation of usability and workload

In the context of using assistive systems, it is of central importance to assess user acceptance within the application domain. For this reason, a user study is conducted to evaluate the usability and the workload when using the tablet application.

5.2.1. Study design & execution

The user study is closely aligned with a practical use case in which a aircraft cabin monument serves as the test object for a manual visual inspection. In this context, five areas with specific defects have been defined (see Figure 8). To ensure 3D defect localization, an AM is attached to the cabin monument in such a way that a single image can capture both the marker and the defined defect areas, which means that only one overview photo is required. As the aim of the study is not to assess the accuracy of the defect classification and description, each participant received a list with the descriptions of the given defects.

Within the study participants (N=16) with an engineering background are asked to perform a visual inspection of a cabin monument using the application and to complete a questionnaire which includes two common, standardized questionnaires. The System Usability Scale (SUS) [28] is used to assess usability, while the NASA Task Load Index (NASA-TLX) [29] is used to assess workload. Along with the SUS test and the NASA-TLX, the questionnaire for the user study contains further self-defined questions. The participants assess whether they consider the two new main functions - defect segmentation and localization - to be useful. In addition, it is determined whether the participants are already familiar with the application from our previous study [16] in order to enable a comparison between new and recurring participants.

5.2.2. Results

In order to evaluate the **usability** of the tablet app, the SUS questionnaire is used for this work, as in the previous work. As

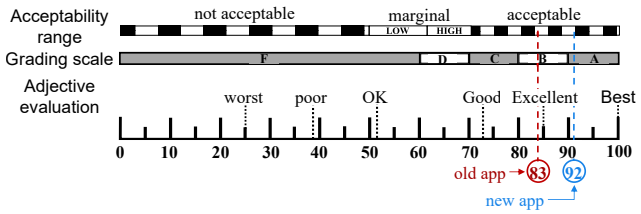


Fig. 7: SUS scale and visualization of the results of the study in this paper and of the comparative value from the previous study [16].

the total score can be between 0 and 100, it is often incorrectly interpreted as a percentage. In order to enable a clearer assessment of usability, the total score determined is converted into a rating scale, which is introduced by Bangor et al. [30]. This enables a more precise and comprehensible evaluation of the app. The evaluation of the SUS test results in an **average total score of 91.875**, which puts the app's user-friendliness in the range "excellent" to "best possible". New users rated the app slightly lower on average with 90.83 points than users who already had experience with the previous version, who rated the app on average with 95 points. In comparison, the user-friendliness of the previous version was rated at 83.25 points in the preliminary study with the identical study design with ten participants [16]. This shows that the introduction of the new functions has resulted in a significant improvement in user-friendliness. Both values are visualized on the SUS scale in Figure 7.

Aspects	∅ Weighting	∅ Value
Temporal Demand	23.33%	25.63
Mental Demand	22.50%	33.75
Frustration	17.50%	15.00
Performance	17.08%	18.75
Effort	10.83%	20.00
Physical Demand	8.75%	18.13

Table 2: Average rating of the individual aspects of the workload sorted by average weighting.

The NASA-TLX questionnaire is applied to assess the **workload** perceived by the participants when using the application to perform the visual inspection. This widely used instrument was originally developed for the aerospace industry and has since been used in the assessment of visual input devices, VR and AR applications as well as automation and decision support systems [29]. The questionnaire is divided into the evaluation and subsequent weighting of six central aspects of workload, which are shown in Table 2. For the interpretation of the determined workload value, the evaluation scale proposed by Prabaswari et al. [31] is used, which enables a systematic classification of the NASA-TLX result. Based on this, the average overall score of **23.33** on the NASA-TLX shows that a **medium workload** is perceived when performing a visual inspection using the assistance system. The weighting of the various workload aspects is particularly noteworthy. While the time and mental demands are perceived as important, the physical demands of carrying out the visual inspection are of secondary importance. Since, in contrast to usability, no values from the preliminary study are available for comparison, the studies by [32] and [33], which examined similar tasks in their studies, are used for contextualization. The results of the system presented are slightly lower than those of the comparative studies, which means that the system appears to have an appropriate workload for the task.

The **additional questions** in the questionnaire on the basic usefulness of the core functions were answered positively by all participants. Furthermore, the participants who had already worked with the previous system are asked to rate the new app in comparison to the old one in terms of the effort required to implement it and its ease of use. Both aspects were perceived as significantly better compared to the old app. Furthermore, additional comments from the participants emphasized that the segmentation tools based on point input (point and foreground/background prompts) are sufficient for the task and that bounding box prompting does not provide any added value for this task. The pictorial documentation of one participant's results in the user study is shown with the overview image and the individual defect images in Figure 8.

6. Discussion

6.1. Discussion of the localization results

The validation method used to determine the accuracy of the localization of defects on the 3D model indicates the presence of a systematic error. In particular, the significant discrepancies between the trueness and precision of the intersection calculations, as well as the significant deviation in the tablet localization accuracy of 26.37 mm, indicate potential shortcomings in the application of the validation systems. Although the specifications of the motion capture system and the robot arm used in their technical documentation show sufficient accuracy, the validation results suggest that the systematic error may be due to a suboptimal configuration or a faulty experimental setup.

The accuracy of the validation depends crucially on the precise alignment of the components used. One challenge is to precisely align the coordinate systems of the OT system and the robot arm. Deviations in this alignment can already be a source of inaccuracies in the determined position of the tablet, but do not directly affect the calculation of the intersection points, as the pose is determined using the AM and its use to generate the intersection lines is independent of the robot arm. The decisive factor here is that the plane in the CAD model is precisely aligned and positioned with the side wall of the test object to which the AM is attached. Although the position and alignment of the side wall can be determined via the OT system, the initial alignment and the definition of the coordinate system in particular harbor potential sources of error that can affect the validation results. A potential inaccuracy in the calculation of the intersection point was already taken into account during the implementation of the overall concept by giving the inspector a selection of affected components that are located near the calculated intersection point in the CAD model. Alternatively, the

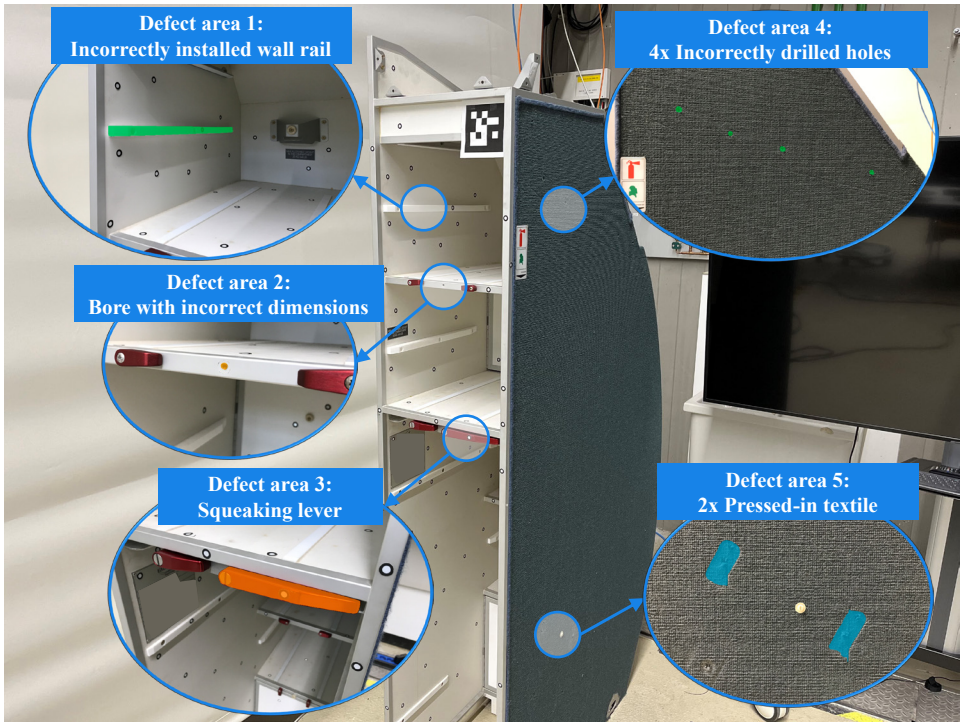


Fig. 8: Defect areas and defects to be segmented within in the user study and exemplary results of one participant.

inspector has the option of manually choosing the affected component. This methodical adjustment ensures that the impacts of a inaccurate localization do not affect the defect allocation to a part and eventually the quality of documentation.

6.2. *Discussion of the user study*

Using the NASA-TLX questionnaire the assistance system demonstrated moderate workload in the application on a use-case of the aircraft industry. However, it should be noted that no direct comparative values are available for alternative documentation methods, such as traditional paper-based or AR-based solutions, for carrying out a visual inspection within this scope of tasks. The same applies to the usability, where an absolute statement can be made about a general very high result, but without any relation to the same performance with other methods. However, studies such as that by Plewan et al. [33], show that tablet applications have higher user acceptance and a lower workload than handwritten approaches. It should also be noted that in this study we have focused on user acceptance as the central object of consideration for human-centered systems. The acceptance of a technology strongly influences process-related key figures [34] such as the error rate and task completion time. These need to be determined in order to further characterize and to demonstrate performance compared to alternative solutions. In addition, the user study was carried out in a laboratory environment with an industrial component and a typical inspection task, but not with the actual potential end users. The significance of the results on user acceptance may be limited.

7. Summary & Future Work

In this work we have developed an assistance system for visual inspection based on the boundary conditions of aircraft cabin inspection. Based on the fundamental tasks and defect classes, which served as a basis, various works on this topic were examined. In particular, the manual inspection of surface defects has so far only been considered in the literature under the premise of existing training data and the goal of automation. The design of an assistance system that can detect different types contours across domains with the help of generalized VFMs contributes to closing this gap. Since this approach also allows the marking of components themselves, the assistance system is in principle also capable of being used beyond the visual inspection of surface defects. The combination with a simple marker-based localization method and the decoupling from the image acquisition of the defect enables defect localization even in areas that are difficult to access. However, the evaluation still showed weaknesses in the area of localization. Inaccurate localization can be taken into account by mirroring the component selection back to the inspector. The user study showed promising results in the area of usability and a clear improvement on the manual segmentation tools of the first system generation. For the workload, values in acceptable ranges were also achieved, which are comparable with related work.

Overall, the system offers significant advantages over current industry practice, which is characterized in particular by the marking of defect areas on paper-based documents. The system follows an end-to-end digital principle, which eliminates the additional effort of digitizing manual notations.

This makes the process more time efficient. In addition, the SAM minimizes the variation of possible segmentation masks across users because the model can be configured to behave deterministically. As a result, we achieved less variation and a reproducible process flow compared to our previous implementation of the assistance system.

Several areas for future research can be identified. First, there is the investigation of alternative, more stable localization options that eliminate the loop back to the user. Marker-based methods can be easily attached to fixtures of large components, but must be visible for the overview photo. An extension to model-based methods, such as those used in augmented reality, and investigating whether these can achieve the desired accuracy for the use case is one aspect.

Within section 4.2 we have already pointed out the weakness of SAM for the task of checking completeness. In this context the use of completely manual segmentation tools leads to inefficient processes. Since this type of check has already been covered several times in the literature, combining SAM for inspection activities with existing components with solutions such as those of [14] and [9] for missing compontents is an approach worth pursuing. A closer look at the developed segmentation tool in an industrial environment is also an important aspect. Involving the actual user group, i.e. inspectors on the shop floor, and getting their feedback is crucial for industrial usability. Furthermore, the correctness of the segmentation was not considered in this work. To evaluate the extent to which the SAM can generate accurate masks in the domain, several components with ground truth masks are required. Metrics such as "intersection over union" can be used here to make an assessment. In this context, the further use of the segmented and annotated data is crucial. The datasets generated with this assistance system can be used to train automated or semi-automated approaches for defect detection. Of particular interest is the tuning of pre-trained networks, as in transfer learning approaches (like in [35], with the generated data. One research question may be to what extent the presented approach can be combined with transfer learning approaches, e.g. to give the user suggestions for detected defects directly in the image when the training of the network has reached sufficient confidence for a specific defect. This approach would combine the strengths of transfer learning approaches with the strengths of human-centered approaches to achieve increasingly accurate visual inspection over time.

CRediT author statement

J. Koch: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing – original draft, Writing - review & editing, Visualization, Supervision, Project administration; D. Jevremovic: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing – original draft, Visualization; K. Moenck: Resources, Writing - review & editing; T. Schüppstuhl: Resources, Funding acquisition, Writing - review & editing.

References

[1] N. Mosca, C. Patruno, V. Reno, M. Nitti, E. Stella, Qualitative comparison of methodologies for detecting surface defects in aircraft interiors, in: 2021 IEEE 8th International Workshop on Metrology for AeroSpace (MetroAeroSpace), IEEE, 2021, pp. 215–220. doi:10.1109/MetroAeroSpace51421.2021.9511778.

[2] J. Koch, G. Lotzing, H. Eschen, K. Moenck, T. Schüppstuhl, A human-centered iiot platform approach for manual inspections: Towards digital documentation and assistance applications, Procedia CIRP 120 (2023) 762–767. doi:10.1016/j.procir.2023.09.072.

[3] K. Schiebold, Zerstörungsfreie Werkstoffprüfung - Sichtprüfung, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. doi:10.1007/978-3-662-44667-6.

[4] Y. D. Yasuda, F. A. Cappabianco, L. E. G. Martins, J. A. Gripp, Aircraft visual inspection: A systematic literature review, Computers in Industry 141 (2022) 103695. doi:10.1016/j.compind.2022.103695.

[5] C. P. Keferstein, M. Marxer, C. Bach, Fertigungsmesstechnik, Springer Fachmedien Wiesbaden, Wiesbaden, 2018. doi:10.1007/978-3-658-17756-0.

[6] N. Mosca, V. Renò, M. Nitti, C. Patruno, E. Stella, Post assembly quality inspection using multimodal sensing in aircraft manufacturing, in: S. Negahdaripour, E. Stella, D. Ceglarek, C. Möller (Eds.), Multimodal Sensing and Artificial Intelligence: Technologies and Applications II, SPIE, 21.06.2021 - 26.06.2021, p. 30. doi:10.1117/12.2594104.

[7] C. V. d. Santos, D. R. Leiva, F. R. Costa, J. A. R. Gregolin, Materials selection for sustainable executive aircraft interiors, Materials Research 19 (2) (2016) 339–352. doi:10.1590/1980-5373-MR-2015-0290.

[8] A. Agarwal, A. Ajith, C. Wen, V. Stryzheus, B. Miller, M. Chen, M. K. Johnson, J. L. Susa Rincon, J. Rosca, W. Yuan, Robotic defect inspection with visual and tactile perception for large-scale components, in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2023, pp. 10110–10116. doi:10.1109/IROS55552.2023.10341590.

[9] J. Hu, G. Zhao, W. Xiao, R. Li, Ar-based deep learning for real-time inspection of cable brackets in aircraft, Robotics and Computer-Integrated Manufacturing 83 (2023) 102574. doi:10.1016/j.rcim.2023.102574.

[10] J. E. See, C. G. Drury, A. Speed, A. Williams, N. Khalandi, The role of visual inspection in the 21 st century, Proceedings of the Human Factors and Ergonomics Society Annual Meeting 61 (1) (2017) 262–266. doi:10.1177/1541931213601548.

[11] C. Deneke, K. Moenck, T. Schueppstuhl, Augmented reality based data improvement for the planning of aircraft cabin conversions, in: Association for Computing Machinery, New York, NY, United States (Ed.), 2021 The 8th International Conference on Industrial Engineering and Applications (Europe) (ICIEA), ACM, New York, NY, USA, 2021, pp. 37–45. doi:10.1145/3463858.3463896.

[12] T. Sheveleva, K. Herrmann, M. L. Wawer, C. Kahra, F. Nürnberger, O. Koepler, I. Mozgova, R. Lachmayer, S. Auer, Ontology-based documentation of quality assurance measures using the example of a visual inspection, in: M. Valle, D. Lehmhus, C. Gianoglio, E. Ragusa, L. Seminara, S. Bosse, A. Ibrahim, K.-D. Thoben (Eds.), Advances in System-Integrated Intelligence, Vol. 546 of Lecture Notes in Networks and Systems, Springer International Publishing, Cham, 2023, pp. 415–424. doi:10.1007/978-3-031-16281-7_39.

[13] R. Hussamadin, G. Jansson, J. Mukkavaara, Digital quality control system—a tool for reliable on-site inspection and documentation, Buildings 13 (2) (2023) 358. doi:10.3390/buildings13020358.

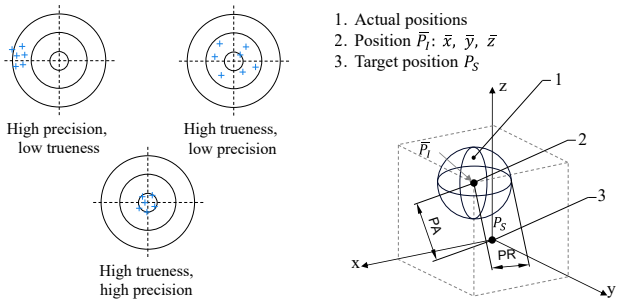
[14] H. Ben Abdallah, I. Jovančević, J.-J. Orteu, L. Brèthes, Automatic inspection of aeronautical mechanical assemblies by matching the 3d cad model and real 2d images, Journal of imaging 5 (10) (2019). doi:10.3390/jimaging5100081.

[15] E. Marino, L. Barbieri, B. Colacino, A. K. Fleri, F. Bruno, An augmented reality inspection tool to support workers in industry 4.0 environments,

Computers in Industry 127 (2021). doi:10.1016/j.compind.2021.103412.
[16] Julian Koch, Denis Jevremovic, Thorsten Schüppstuhl, Mobile web app for the digitization and annotation of manual visual inspection tasks. doi:10.13140/RG.2.2.31427.58402.
[17] S. Piontek, M. Schütze, H. Lödding, Digital assistance for aircraft manufacturing – process requirements and technologies, Procedia CIRP 120 (2023) 105–110. doi:10.1016/j.procir.2023.08.019.
[18] L. Büsch, J. Koch, D. Schoepflin, M. Schulze, T. Schüppstuhl, Towards recognition of human actions in collaborative tasks with robots: Extending action recognition with tool recognition methods, Sensors (Basel, Switzerland) 23 (12) (2023). doi:10.3390/s23125718.
[19] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, R. Girshick, Segment anythingdoi:10.48550/arXiv.2304.02643.
[20] K. Moenck, A. Wendt, P. Prünte, J. Koch, A. Sahrhage, J. Gierecker, O. Schmedemann, F. Kähler, D. Holst, M. Gomse, T. Schüppstuhl, D. Schoepflin, Industrial segment anything – a case study in aircraft manufacturing, intralogistics, maintenance, repair, and overhaul (2023). arXiv:2307.12674.
[21] Meta Open Source, React: The library for web and native user interfaces, [Online] (Zugriff am: 21.02.2024).
URL https://react.dev/
[22] Material UI SAS, Mui: Move faster with intuitive react ui tools, [Online] (Zugriff am: 21.02.2024).
URL https://mui.com/
[23] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, M. J. Marín-Jiménez, Automatic generation and detection of highly reliable fiducial markers under occlusion, Pattern Recognition 47 (6) (2014) 2280–2292. doi:10.1016/j.patcog.2014.01.005.
[24] OPEN CASCADE SAS, pythonocc provides a python wrapper for the opencascade c++ technology., [Online] (Zugriff am: 26.02.2024).
URL https://dev.opencascade.org/project/pythonocc
[25] Universal Robots, Ur10e: Medium-sized, versatile cobot, [Online] (Zugriff am: 17.02.2024).
URL https://www.universal-robots.com/products/ur10-robot/
[26] NaturalPoint Corporation, Optitrack for robotics, [Online] (Zugriff am: 22.12.2023).
URL https://optitrack.com/applications/robotics/
[27] ISO Internationale Organisation für Normung, Industrieroboter - leistungskenngrößen und zugehörige prüfmethoden (1998).
[28] J. Brooke, Sus: A 'quick and dirty' usability scale, in: P. W. Jordan, B. Thomas, B. A. Weerdmeester, I. L. McClelland (Eds.), Usability evaluation in industry, Taylor and Francis, Bristol, PA and London and ©1996, 1996.
[29] S. G. Hart, Nasa-task load index (nasa-tlx); 20 years later, Proceedings of the Human Factors and Ergonomics Society Annual Meeting 50 (9) (2006) 904–908. doi:10.1177/154193120605000909.
[30] A. Bangor, P. T. Kortum, J. T. Miller, Determining what individual sus scores mean: adding an adjective rating scale, Journal of Usability Studies archive 4 (2009) 114–123.
URL https://api.semanticscholar.org/CorpusID:7812093
[31] A. D. Prabaswari, C. Basumerda, B. W. Utomo, The mental workload analysis of staff in study program of private educational organization, IOP Conference Series: Materials Science and Engineering 528 (1) (2019) 012018. doi:10.1088/1757-899X/528/1/012018.
[32] C. Liebers, M. Prochazka, N. Pfützenreuter, J. Liebers, J. Auda, U. Gruenefeld, S. Schneegass, Pointing it out! comparing manual segmentation of 3d point clouds between desktop, tablet, and virtual reality, International Journal of Human–Computer Interaction (2023) 1–15doi:10.1080/10447318.2023.2238945.
[33] T. Plewan, B. Mättig, V. Kretschmer, G. Rinkenauer, Exploring the benefits and limitations of augmented reality for palletization, Applied ergonomics 90 (2021) 103250. doi:10.1016/j.apergo.2020.103250.
[34] L. Mlekus, D. Bentler, A. Paruzel, A.-L. Kato-Beiderwieden, G. W. Maier, How to raise technology acceptance: user experience characteristics as technology-inherent determinants, Gruppe. Interaktion. Organisation. Zeitschrift für Angewandte Organisationspsychologie (GIO) 51 (3) (2020) 273–283. doi:10.1007/s11612-020-00529-7.
[35] Y. Gong, H. Shao, J. Luo, Z. Li, A deep transfer learning model for inclusion defect detection of aeronautics composite materials, Composite Structures 252 (2020) 112681. doi:10.1016/j.compstruct.2020.112681.
[36] B. Priemer, Unsicherheiten, aber sicher!, Springer Berlin Heidelberg, Berlin, Heidelberg, 2022. doi:10.1007/978-3-662-63990-0.

Appendix A. Accuarcy, trueness and precision

In metrology, accuracy is defined as the result of precision and trueness. Precision describes how close measured values are to each other in repeated measurements, while trueness describes how close a measured value is to a specified reference value (see Figure A.9a) [36]. The trueness can be calculated as the Euclidean distance between the measured value and the reference value (see Eq. A.1), which is defined in EN ISO 9238 as the positional accuracy (PA). To determine the accuracy, the average of all the measured values from the repeated measurements is required to determine the scatter of all the measured values. The equation A.2 can be used to calculate a parameter for precision, which is referred to in ISO 9238 as position repeatability (PR). This is depicted in Figure A.9b.



(a) Comprehension of accuracy in metrology.

(b) Representation of the accuracy parameters PA and PR.

Fig. A.9: (a) Description of the term "accuracy" and (b) Presentation of the most important variables for determining position accuracy (PA) and position repeatability (PR) (based on [27, 36]).

$$PA = \sqrt{(\bar{x} - x_s)^2 + (\bar{y} - y_s)^2 + (\bar{z} - z_s)^2}$$
$$\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j; \quad \bar{y} = \frac{1}{n}\sum_{j=1}^{n} y_j; \quad \bar{z} = \frac{1}{n}\sum_{j=1}^{n} z_j; \tag{A.1}$$

$$PR = \bar{l} + 3S$$
$$\bar{l} = \frac{1}{n}\sum_{j=1}^{n} l_j$$
$$l_j = \sqrt{(x_j - \bar{x})^2 + (y_j - \bar{y})^2 + (z_j - \bar{z})^2} \tag{A.2}$$
$$S = \sqrt{\frac{\sum_{j=1}^{n}(l_j - \bar{l})^2}{n-1}}$$