AI+AR based Framework for Guided Visual Equipment Diagnosis

Teresa Gonzalez Diaz[1], Xian Yeow Lee[1], Huimin Zhuge[1], Lasitha Vidyaratne[1], Gregory Sin[1], Tsubasa Watanabe[2], Ahmed Farahat[1] , Chetan Gupta[1]
[1] Hitachi R&D America, Santa Clara, Ca, 95054, USA
teresa.gonzalezdiaz@hal.hitachi.com
xian.lee@hal.hitachi.com
joy.zhuge@hal.hitachi.com
lasitha.vidyaratne@hal.hitachi.com
gregory.sin@hal.hitachi.com
ahmed.faharat@hal.hitachi.com
chetan.gupta@hal.hitachi.com
[2] Hitachi R&D America, Holland, Mi, 49424, USA
tsubata.watanabe@hal.hitachi.com

ABSTRACT

Automated solutions for effective support services, such as failure diagnosis and repair, are crucial to keep customer satisfaction and loyalty. However, providing consistent, high quality, and timely support is a difficult task. In practice, customer support usually requires technicians to perform onsite diagnosis, but service quality is often adversely affected by limited expert technicians, high turnover, and minimal automated tools. To address these challenges, we present a novel solution framework for aiding technicians in performing visual equipment diagnosis. We envision a workflow where the technician reports a failure and prompts the system to automatically generate a diagnostic plan that includes parts, areas of interest, and necessary tasks. The plan is used to guide the technician with augmented reality (AR), while a perception module analyzes and tracks the technician's actions to recommend next steps. Our framework consists of three components: planning, tracking, and guiding. The planning component automates the creation of a diagnostic plan by querying a knowledge graph (KG). We propose to leverage Large Language Models (LLMs) for the construction of the KG to accelerate the extraction process of parts, tasks, and relations from manuals. The tracking component enhances 3D detections by using perception sensors with a 2D nested object detection model. Finally, the guiding component reduces process complexity for technicians by combining 2D models and AR interactions. To validate the framework, we performed multiple studies to:1) determine an effective prompt method for the LLM to construct the KG; 2) demonstrate benefits of our 2D nested object model combined with AR model.

1. INTRODUCTION

Offering support services has become a key differentiator for customer satisfaction and retention in multiple industries. For example, manufacturers provide products along with support services and warranties to ensure that machines' downtime is minimized. However, operational complexities hinder the overall quality of services, such as limited experienced technicians, high turnover, steep learning curves of the manuals and few automated tools. Therefore, it is essential to develop automated methods and systems for technicians' assistance that aim to high standards of support services.

Building intelligent assistant systems present important technical challenges. First, knowledge bases are required to provide reasoning and extensibility, but traditional methods require extensive data and labels. Second, scene understanding is critical to guarantee the quality of visual guidance, but existing methods are not sufficient for environment variations of customer sites. Third, advanced user interfaces are required to be intuitive and useful, but Augmented Reality (AR) with 3D methods, though enabling rich human interactions, are slow with limited generalization.

To tackle these challenges, we proposed a novel general framework for guided visual diagnosis. In our approach, the system assists technicians in their tasks, irrespective of their experience level and the complexity of the issues they encounter. The framework integrates methods that facilitate an automated, interactive and user-friendly approach.

In summary, our approach comprises the following contributions:

1. A novel general framework designed to automate the visual diagnosis process enabled by methods for diagnostic plan generation, tracking and AR guidance.

2. A method to automatically generate a diagnostic plan by querying a knowledge graph (KG). Our KG is created by leveraging LLMs for entity extraction of *equipment parts and diagnosis tasks* from unstructured sources. The KG also includes spatial entity relationship of *diagnosis area* extracted from images annotations.
3. A method to automatically track and guide the diagnostic state using inputs from perception sensors in real-world environment. Our method uses a 2D object detection model combined with AR-based 3D positioning that overcomes intrinsic equipment challenges such as highly reflective areas, translucid, obstacles or occlusion.

The paper is structured as follows: Section 2 describes related work on methods, technology and trends to develop guidance systems. Section 3 describes problem and definitions, followed by methodology and method details in Section 4 and 5. Section 6 discusses the experiments and results. Finally, Section 7 concludes the paper and discusses future work.

2. RELATED WORK

Industrial research has been influenced by the proven success of AI-based computer vision and Augmented Reality (AR) in other domains. AI-based methods have been proposed for industrial problems such as defect detection, quality control, repair recommendations, etc. Limited assistants and service robots' systems have been proposed to facilitate technician's jobs. With the recent breakthrough of large language models (LLMs), new research have emerged to improve assistance technology. Related work is summarized as follows.

2.1. Assistant Systems

Automated support systems include two types of assistant tools: 1) remote assistance with video calls, and 2) AI-based chatbots [1]. Remote systems commonly require a human expert with limited interactivity, coordination, and collaboration. Chatbots have been widely used for text-based search on manuals or databases. Others, with more user interaction such as animations or simulations, are usually built for specific cases and products using databases, custom rules and 3D-CAD models. In [2], the authors proposed a visual assistant for inspection where technicians are guided to inspect a vehicle and detect surface defects. However, to our knowledge, no work has been done on visual assistance systems for diagnosis. Different from remote and chatbots solutions, our framework enables a near real-time visual-based system to assist technicians on root cause analysis.

2.2. AI-based Visual Systems

AI-based visual systems have been widely incorporated to analyze images for defects detections in different types of equipment analysis. In [2,3,4], authors demonstrated that deep learning-based methods have an acceptable performance in applications such as crack detection in roads, welding, buildings, etc.; and surface damages in railroads or vehicles. Common methods used in these systems are 1) classifiers such as ResNet, MobileNet, and Swin Transformers; 2) real time object detectors such as theYolo series, 3) instance segmentation models such as MaskRCNN, Yolo, and Segment Anything Model (SAM) [17,18]. Minimal work has been done on visual root cause detection.

2.3. Knowledge Graphs and Entity Extraction

Advanced assistant systems generally include a form of Knowledge Base (KB) to draw information required for a wide range of assistive tasks. Knowledge bases (KB) and graphs (KG) have been studied extensively. Traditional approaches still require large amounts of data that make KB development process complex and slow. The main task of building knowledge bases and graphs is entities and relation extraction (RE) that enables reasoning based on the graph semantics. Popular approaches to extract entities and relationships include custom seq2seq models [1] and REBEL [6]. However, with advancements in LLMs, such as ChatGPT [7], entity tagging, and relation extraction has been re-evaluated to assess their potential performance for domain-specific knowledge [8]. In [9,10], authors demonstrated that it is possible to achieve SOTA performance on relation extraction with minimal training data. In our case, we study the viability of using LLMs to extract industrial entities such as parts and diagnosis tasks with minimal data.

2.4. Large Language Models

Large language Models (LLMs) have been recently incorporated in chatbot-based systems and other downstream tasks such text classification, summarization, entity extraction, etc. [10]. LLMs have been trained on massive datasets for language reasoning and generation based on Transformer architectures. Multiple LLMs have become popular such as ChatGPT from OpenAI, Llama2 from Meta Research, Gemini from Google Research [18]. Training these models for a specific domain or task may be infeasible due to the amount of required data and hardware resources to achieve high performance. However, utilizing fine-tuning techniques, the models can be easily adapted with few samples. For example, Prompt Fine-tuning is a popular technique which employs prompts to instruct the pre-trained model to understand the new context. As tailoring process, Prompt Elicitation facilitates the definition of the prompts, main model role, task, and response requirements. Examples of prompt types for fine-tuning are text completion, task summarization, information retrieval, question-answering.

2.5. Scene Understanding

Substantial research has been done on scene understanding, especially in areas like autonomous driving systems and service robots. Scene understanding approaches analyze and interpret the functional context and semantics of objects with

respect to the relationship with the 3D space and physical layout. According to [11,12], scene understanding methods solve problems like reasoning, finding hidden objects and completing objects. Understanding the scene structure is categorized into 1) object-oriented and 2) spatial-oriented. For instance, in [3], a relevant scene graph representation is proposed to captures the objects and their relationships on the layout, e.g. rooms, garden, etc. However, traditional 2D and 3D approaches struggle with low accuracy and stability in scenarios involving transparent objects and high reflections. In contrast, our approach overcomes transparency and reflections by targeting well-defined key objects (parts) and their related objects of interest (parts/components) contained within a spatial layout as areas of interest.

2.6. Augmented Reality Interaction

With the proliferation of Augmented Reality (AR) libraries like Apple ARkit, Google ARCore, and WebAR, several AR approaches have been designed for navigation guidance, assembly tracking, and repair assistant[11]. However, there are minimal visual guidance for diagnosis. According to [13], most of the AR cases integrate overlay annotations to interact with the user. AR applications overlay the annotations using either 1) physical markers (for example lines over floor, or bar codes over objects) or 2) 3D object recognition to identify the target objects. The identification process involves three steps. First, a 3D scanner registers the environment. Second, 3D object representations are generated, such as CAD objects, point clouds, etc. Third, an application uses these 3D representations to recognize the scanned objects. However, the accuracy of 3D model is negatively affected by varying environments (background, area, lighting, layout, etc.). In practice, creating a comprehensive 3D environment is difficult and sometimes infeasible. In addition, 3D recognition is still a challenge for real time systems (i.e. under one second) [11,3]. Different from these approaches, we propose 2D object recognition combined with 3D positioning model that outperform traditional 3D detection.

3. PROBLEM DEFINITION

We define a *visual equipment diagnosis VED* as the process required to find the root cause of a failure *f* for a given equipment *E*. For example, "the refrigerator's freezer is making a lot of noise" where "refrigerator" denotes *E* and "making a lot of noise" denotes *f*. Since the failure *f* may often represent only a possible symptom of one or multiple underlying issues, the technician needs to conduct a comprehensive evaluation. For example, to identify the root cause of the freezer noise, the technician may need to check multiple parts such as the freezer, thermostat, motor, etc. In practice, based on experience or searching in equipment documents, the technician follows a sequence of steps *S* to diagnose the root cause. The steps involve troubleshooting tasks on related equipment parts. For example, as a first step, "check freezer interiors", second "turn on and off the thermostat", and third "check the motor".

3.1. Definitions

Formally, we assume that the technician follows a **diagnostic plan *DP*** to identify the root cause of a failure *f. DP* is defined as the list of steps *S* that involves parts *P* and tasks *T*. Let *P* be a list of parts related to the failure such as $f \rightarrow P$. Let *T* be diagnostic tasks denoted as actions performed on *P* to determine a root cause such as $T \rightarrow f$. Specifically, the following definitions are considered:

- **Diagnostic Area**: Let equipment *E* with 3D structure be composed of viewpoints $V=\{v_1,...,v_n\}$ *where* $n>0$. *V* denotes spatial planes of the equipment and is used for physical navigation. For example: front, back, side, etc. Let a viewpoint v_i, be composed by areas $A=\{a_{1,i}...,a_{m,n}\}$ *where* $i>0$, $m>0$ *and* $n>0$. *A* denotes a set of mutually exclusive splits within the viewpoint v_i. For example, top, and bottom are possible areas of front viewpoint.
- **Diagnostic Part.** Let $P=\{p_1,...,p_k\}$ *where* $k>0$ be the list of parts *such as* p_1 is visible from one or more areas $a_{j,i}$. e.g. filter at front-top, motor at back-bottom.
- **Diagnostic Task**. Let $T=\{t_1,...,t_k\}$ *where* $k>0$. be the list of tasks to be performed by technician at the part p_i *T* are troubleshooting actions that aim to identify the root cause. For example: verify temperature, turn on the switch, check the LED light, etc.
- **Diagnostic Step**. Let $S=\{s_1,...,s_k\}$ *where* $k>0$ be a list of steps to be completed during the diagnosis. Each step s_i is defined as $s_i = (p_i, t_i)$ where part p_i is the part to check and the t_i the task to perform. For example: step 1 is (thermostat, verify temperature), step 2 (switch, change temperature to 20 degrees), etc.
- **Diagnostic Requirements**. Let $R=\{r_1,..., r_k\}$ *where* $k>0$ be denoted as the expected *requirement* r_i of the visual observation to be satisfied at step s_i, and part p_i. For example: size, coverage, color, orientation, etc.
- **Guidance Action:** Let $G=\{g_1,... g_l\}$ *where* $l>0$ be a list of multi-modal messages (visual, text, voice, etc.) that technicians need to follow. *G* is a function of the sampled observation θ at time *l* and its evaluation for a given step s_j and requirements r_i. For example, technician is guided by text messages like "move closer", "open door", etc.; and AR indicators like overlays, animations, etc.

3.2. Problem Overview

Herein, given a failure *f* reported on an equipment *E,* the *guidance problem for a* visual equipment diagnosis *GVED*, is defined by the following two subproblems:

1) Determine a diagnostic plan *DP* from unstructured sources e.g. manuals, reports. *DP* denotes a sequence of

steps S to diagnose parts P with diagnostic tasks T. Each part p_i is physically located and visible at least one area $a_{j,i}$ within at least one viewpoint v_i. See Eq. 1.

$$DP = \{s_1, \ldots, s_k\}$$

$$\textit{Where} \quad s_i = (p_i, t_i) \tag{1}$$

$p_i \subset area\ a_{j,i} \subset viewpoint\ v_i;\ k>0,\ i>0\ and\ j>0.$

2) Determine a *diagnostic state DS* from observations of the *DP* executed by the technician. *DS* is denoted as a function of the guidance G for a given observation θ_l of the step s_i whose outcome determines the expected state such as $G \rightarrow DS$. Let observations θ_l be inputs from a variety of sources (visual, position, motion, etc.) that represent the state of step s_i or transition to s_{i+1}. See Eq.2.

$$DS(\theta_l, s_i) = \begin{cases} G(\theta_l, s_{i+1}, r_{i+1})\ if\ s_i\ is\ completed \\ G(\theta_l, s_i, r_i)\ otherwise \end{cases} \tag{2}$$

Where

$$\theta_l = \{image,\ motion,\ pose\}$$

$\forall\, i>0$ *is the step and* $l>0$ *is the sample time.*

3.3. Constraints

1. **Diagnostic Plan *DP*** generation is subject to uncertain content of diagnosis and troubleshooting details. For example, manuals or reports may include variations or limited information of parts, task, causes, solutions, etc.
2. **Diagnostic State *DS*** detection is subject to four main constraints. First, visual equipment conditions may be uncertain due to encounter with intricate conditions due to installation field (factories, stores, etc.) and day-to-day usage. Second, industrial equipment usually has translucid materials e.g. glass or reflective surfaces e.g. aluminum. Third, technician's behavior and system state are uncertain due to human reasoning, actions and motions. Fourth, state detection requires fast response times for effective user interaction (less than a second).

In summary, we propose a solution that involves: 1) generation of a diagnostic plan *DP* from documents (manuals, reports, etc.) stated in Eq.1.; and 2) detection of the diagnostic state *DS* from perception inputs (video, motion, pose) to track and guide technician actions stated in Eq.2.

4. METHODOLOGY

We propose a general framework that enables automated diagnosis of equipment and ensures quality of visual diagnostic from a non-expert technician. To assist the technician during the process, our approach automates three mechanisms:1) generate a diagnostic plan, 2) track technician actions and motion; 3) guide the technician to troubleshoot the related parts that are possibly causing the failure.

4.1. Guided Visual Diagnosis Framework Overview

Figure 1 illustrates our approach that comprises three main components: planning, tracking and guidance. The summary is described as follows:

1. **Planning**: To improve the slow process (e.g. days, weeks) of manual work needed to construct diagnostic plans for a large list of possible failures, we proposed an automated method to generate them in minutes. Figure 1-a shows the method that automatically builds the diagnosis plan by querying a knowledge-graph (KG) constructed by leveraging LLMs. To generate the diagnostic plan *DP*, the knowledge-graph is traversed to find the node with the reported failure and enumerate a sorted list of steps using its related neighbors (parts, areas, viewpoint, and tasks). The output plan with parts and task is sent to the diagnostic tracking process.
2. **Tracking**: To accurately assist and track the technician's progress at the field, e.g. customer store, factory, etc.; we proposed a multi-modal scene understanding method. Figure 1-b shows our method that combines 2D object detection with 3D perception inputs e.g. camera and motion sensors. The method analyzes the sensor inputs to detect semantics of spatial layout with respect to parts *P* and requirements *R* to evaluate i.e. viewpoints, areas of interest, size, orientation, position and coverage. The combination of the 2D nested object with 3D position readings ensure the correctness of the observation e.g. distance to the equipment, orientation, etc. The detections are sent to the AR Guidance component.
3. **Guidance**: To enrich technicians' experience and reduce complexity, time and number of visits required for diagnosis services, we proposed AR-based guidance. To make the actions intuitive for technicians, Figure 1-c shows that our component utilizes AR indicators according to the progress of the diagnostic plan *DP* using icon overlays, edge alignment, navigation indicators.

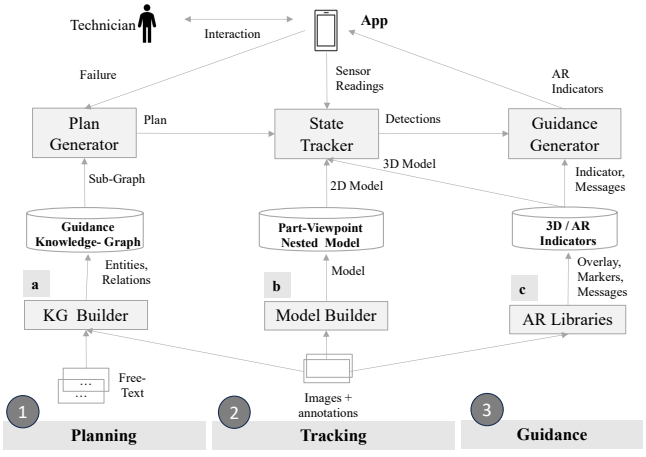

Figure 1. General Framework for Visual Diagnosis Guidance based on AI and AR Methods.

4.2. Guided Diagnosis Use case

Our vision of a guided diagnosis system is to enable effective maintenance and repair of complex equipment such as industrial machinery for medical devices, energy, manufacturing, etc. regardless of technicians' domain experience. Industrial machinery is commonly designed with complex components including high reflective and translucid areas as shown in Figure 2. The assistant system assumes a technician using a mobile device e.g. a phone or tablet. The proposed process consists of three steps. First, the technician informs the failure to diagnose. Then, the system automatically generates the diagnostic plan *DP* to be performed. Second, the camera live feed and AR environment are activated, and 2D/3D detector and AR tracking also start running to guide the technician. Third, the system guides the technician to follow the steps in the plan. Perception sensors are sampled *l* times per second and compared to the requirements *R*. Depending on the evaluation of the observed conditions θ_i, the system generates guidance actions e.g. move to…, go closer, etc. to find the area-part of s_i or continue to next step s_{i+1}. If the area-part is successfully found, it is recorded automatically. The process terminates once all steps are completed. Figure 3 illustrates the use case.

5. METHODS IN DETAIL

5.1. Plan Generation Method

The task is to determine the diagnostic plan *DP* that the technicians need to follow for a given failure *f*. The method automatically builds the diagnosis plan using two components: a) Knowledge Graph Builder (KG) by leveraging LLMs and b) Plan Generator by querying the constructed graph. Figure 4 shows the proposed method, and the details are explained in the following sections.

5.1.1. Knowledge-Graph Builder

To construct a knowledge graph, the component uses free text from existing manuals, and reports where elements are examined when a problem is reported. To achieve this, we 1) leveraged ontology design to define main concepts for building the KG, 2) used a LLM to extract part-task entities from documents, and 3) developed an algorithm to extract spatial entities from image annotations.

Figure 2. Example of Industrial Machinery for Services like Medical, Food, Consumer, Electronics, etc. [19]

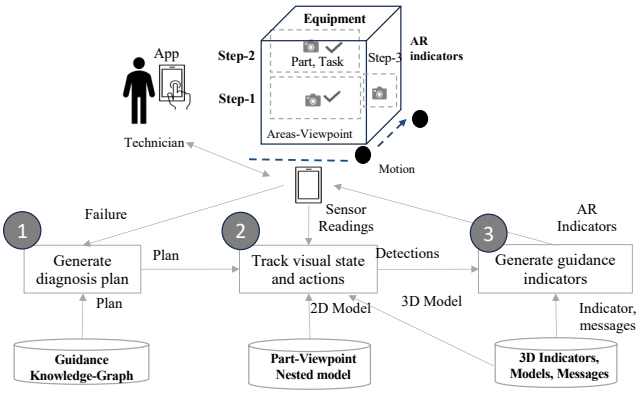

Figure 3. Example of the proposed Use Case for Guided Visual Equipment Diagnosis

a. **Ontology design:** Our ontology enables reasoning with categories of *{parts P, tasks T, areas A, viewpoints V, Failures F}*. *{Parts}* are categories automatically extracted from text using a LLM described below. *{Viewpoint, area and part relationship}* are extracted from the image annotation dataset described below. Figure 4-a1 shows the classes and relations required to create the *diagnostic plan DP* for the failure *f*.

b. **LLM-based Entity Extraction:** To automate the generation of a diagnosis plan composed by steps $s_i = (p_i, t_i)$ with *parts* p_i and *tasks* t_i, we proposed a relation extraction (RE) model. In this context, entities represent parts p_i, while the relationship represents diagnosis task t_i to be performed at the step s_i. To avoid expensive model training or finetuning, we apply a prompt fine-tuning technique for ChatGPT model as completion task. In [8], authors have demonstrated that few-shot prompting with proper instructions and examples can achieve SOTA performance. This method reduces time of manual text extractions from weeks to minutes.

c. **Spatial Entity Extraction**: To remove the manual process of mapping the part to the spatial plane i.e. viewpoint, area, with a mapping algorithm: $P \subset (V,A)$. The method exploits an image dataset with annotations as follows: Each image contains annotations related to the found parts *P*, areas A and viewpoints *V*. First, we extract the annotations, and the portion of the images related. Then, we compute the intersection of union (IoU) to automatically determine the area and viewpoint of the part. Finally, we use the extracted triplets (viewpoint, area, part) to add to the knowledge graph as seen in Figure 4-a3.

5.1.2. Plan Generator

Once a failure *f* is reported, the method queries the KG to traverse the graph with related parts, tasks and areas to examine. The plan generator constructs a list of steps *S* to complete and the relationship with viewpoint + area, part, list of requirements to satisfy such size, and orientation.

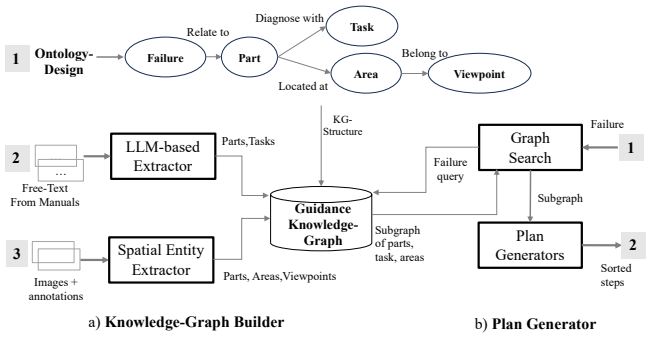

Figure 4. Functional Design for Diagnostic Planning Component.

The list is sorted by viewpoint to minimize the movement during the diagnosis. Figure 4.b shows the plan generator modules.

5.2. Diagnostic State Detection Method

With the goal of understanding the diagnosis state performed by the technicians, we proposed scene understanding methods that uses perception sensors (camera and motion) to determine the visible objects, and their size, orientation, position, and the relation with respect to the checklist plan. To achieve this, the component uses 2 main modules: 2D area detection (Nested Object Detection) and 3D positioning. Figure 5 shows the diagnosis tracking modules and flow.

5.2.1. Nested Object Detection

The task involves detecting spatial information to assess the diagnostic state with respect to the technician, equipment, and environment including the viewpoint (e.g. top, middle, etc.) and areas of interest (e.g. controller, refrigeration, etc.) to initiate checklist AR guidance. Traditional object detectors should work off-the-shelf; however, certain characteristics of the machines, such translucid areas, reflection, and uncertain conditions, hinder achieving good accuracy with these models.

We proposed a nested object detector to boost detection accuracy, especially for translucid or high reflective surfaces in which conventional detectors fail. We defined nested annotations of parts, areas, and viewpoints with the following definitions:

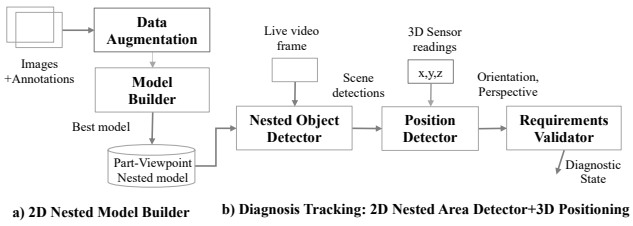

Figure 5. Functional Design for Diagnostic Tracking Component.

a. Definitions

- **Spatial Objects (Viewpoint+Areas)**: Defined as spatial identification of the object (part) with respect to the 3D planes of the equipment viewpoints *V* and areas *A*. The detector is trained with labels such as {(front, top), (front, middle)}.
- **Parts**: Areas of interest that the technician needs to evaluate. For example, filter, container, handle, etc.
- **Key Anchor Objects**: We named anchor objects *(KaO)* as those objects well defined by shape, contrast, light, color, etc. Key anchor objects include a set of 1 or more objects nested on the target viewpoints-areas, where $KaO_{j,i} \in A \in V$, i>1 and j>1. The key anchor objects are selected to propagate loss function activation for area and viewpoint classes.
- **Spatial relationships:** We use viewpoint + area detections to localize the part and guide the technician.

Figure 6 displays the example of the object definitions above. We show the areas of interest: top, middle (mid) and bottom for a viewpoint front: front-top, front-mid, front-bottom.

a. **Training:** We selected and labeled visual semantics with nested objects (parts, key anchor objects-parts) and outer target objects (viewpoint + area). Our task is to detect the viewpoints-area for navigation and parts for diagnosis.

a. **Inference**: During the diagnosis, we run the inference every three frames per second. We select detections with confidence score greater than 0.5.

- To improve viewpoint-area detection, anchor objects are used to imply and inhere confidence score where viewpoint-areas are compromised by reflection or translucid areas.
- The detected bounding boxes are utilized to determine the relative size, 2D position and coverage within the scene. We leverage the result of object detection (bounding boxes of object of interest) to calculate object size within the image frame and estimate their expected area coverage.
- Finally, to determine the 2D spatial semantics of the object, we used objects localization and orientations in the scene to validate requirements.

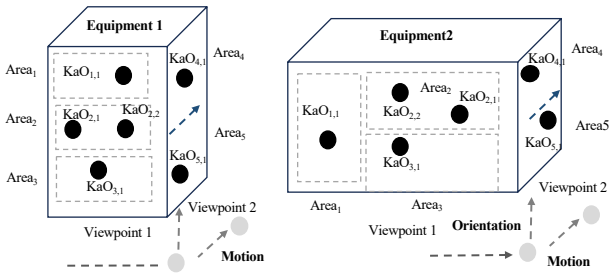

Figure 6. Example of Nested Objects for Boosting Activation of Areas and Viewpoint Classes.

5.2.2. 3D Positioning Detection

As presented in Figure 5-b, 3D Positioning method utilizes motion sensor readings to determine the orientation and perspective in terms of roll and pitch. The proposed method leverages the accelerometer and gyroscope sensors available on handheld devices. The goal of this module is to detect the orientation and alignment of the camera relative to the area of interest (i.e. while holding the handheld device). Sensor readings are integrated with the object detection outcomes (bounding boxes of object of interest) to determine distance to the object, its size, orientation and whether it is centered in the frame. For example, in an area of interest, valid sensor readings should indicate a 90-degree vertical alignment view, allowing orientation markers to indicate if the camera's orientation is correct or not.

6. EXPERIMENTS AND RESULTS

We performed experiments to study the feasibility of our proposed framework in the key methods: 1) plan generation for diagnosis, and 2) state detection for tracking and guiding.

6.1. Plan Generation Method

To evaluate the performance of the generated plan for diagnosis, we did a study of the entity relation (ER) extraction using LLM. Entity extraction of parts and tasks is the core component to construct the knowledge graph and directly represents the performance that can be achieved for the generation. Once the extraction is done, we assume that querying the graph is straightforward. We emphasized extraction performance to gauge the effectiveness of the plan generation. Traditional approaches required extensive datasets to achieve acceptable performance in ER problems as mentioned in Section 2. Therefore, our goal is to measure achievable performance using small datasets while leveraging LLM.

6.1.1. Dataset settings

We used troubleshooting sections from publicly available manuals of two types of equipment: smart refrigerator [14], and vending machine [15]. The troubleshooting descriptions were extracted from their manuals which commonly specifies parts related to the failure and tasks to identify the root cause. We constructed a testing dataset of 100 records for each equipment. Figure 7 shows a snapshot of a type of equipment and descriptions used. Additionally, we used private industrial machinery datasets to evaluate our methods. However, due to confidentiality restrictions, we have only reported public datasets and results.

6.1.2. Experiment Settings

We prepared 3 types of experiments using a LLM model.

Equipment: Refrigerator
Failure: Cooling systems is running to much.
Troubleshooting Description: The doors are opened, or food containers are obstructed areas. Verify doors alignment and containers organization. The amount of food or warm food required the compressor to run longer. Verify food temperature volume in containers. Test thermostat operations.

Equipment: Vending machine
Failure: Coin mechanist not accepting coins.
Troubleshooting Description: Coin mechanism harness to control board is cut or disconnected. Check changer harness. Shor in coin mechanism. Unplug all harnessing from control board and test coin mechanism connection acceptance, if accept, re-plug and test acceptance again.

Figure 7. Examples of Troubleshooting Description for a Refrigerator and Vending Machine.

Specifically, given the known performance and popularity of ChatGPT, we evaluate GPT 4.0 models as a completion task with different prompting approaches. The goal is to study the minimum data requirements that maximize the precision of entity relation extraction. To evaluate that, the experiment design consists of prompt elicitation as follows:

1. **One-shot prompt (P-1S)**. Prompts designed to provide a troubleshooting description from manuals. We instruct the system to extract parts and task related to the description.
2. **One-shot prompt with example (P-1SE)**. Prompts designed to provide a troubleshooting description from manuals. We instruct the system to extract parts and task related to the description along with an example of parts and task extracted from the given description.

Additionally, we also analyzed the impact of extended context. To do this, prompts are designed in two ways:

a. **Specific context**: For each description record, we instantiate a new a session with ChatGPT which ensures no previous context of our manuals can be related during the model completion.
b. **Extended context**: We instantiate a single session for the testing dataset which facilitates model completion to use previous context of our manuals.

Experiments with extended context prompt with the postfix "+" such as **P-1S+ and P-1SE+.** For example, P-1S+ refers to the experiment with one-shot prompt with extended context. Figure 7 shows examples of the prompt descriptions.

6.1.3. Evaluation Method

Evaluating ChatGPT results is complicated due to hallucinations and semantically similar response with different words. To overcome this challenge, we used ChatGPT as an evaluator instructed with a classifier prompt. The role was to determine if the extraction is correct or not (Yes/No). If the extraction was incorrect, ChatGPT was asked to determine if the extraction may be *related, implied or inferred* from the text with Yes/No response. If the subsequent response was Yes, we assumed the extraction result was a hallucination, otherwise we treat the extraction as incorrect.

6.1.4. Evaluation Metrics

We used 3 main metrics:

1. **Precision**: True Positives are defined as extractions that were classified correctly as mentioned in the description. False Positive are defined as extractions that were not mentioned in the description.
2. **Hallucination Rate** (HAL-R): Hallucination is defined as the extraction that could be implied, referred or related as a potentially correct answer even though it is not explicit in the description.
3. **Precision + HAL-R**: We redefined True Positive as the traditional true positive that also accounts for the correct hallucinations, defined as TP = TP + HAL. False Positives are extractions that are neither mentioned and are not hallucinations.

6.1.5. Results

The results in Tables [1,2] show that our proposed KG construction method is feasible. Specifically, precision is above 92.83% for part extraction and 84.59% in tasks. When considering hallucination as true positives, then Precision+HAL is at least 95.34% for parts and 92.74% in tasks. Figure 8 shows examples of the input and output obtained in the experiments. Results summary is as follows:

1. **Prompt Comparison**: Based on the three types of prompts studied, the results show:
   - **One shot result**: Providing instructions helps but providing a relevant example along with the instruction boost the precision. For example, a maximum gain of 18.02% was achieved between P-1S+ and P-1SE+ of refrigerator experiments. Overall, all the experiments with instruction and example tuning increased their precision and reduced hallucination.
   - **Extended context results**: Including extended context (P-1S+ and P-1SE+) does not exhibit any consistent pattern. We hypothesize that this is because previous descriptions do not necessarily have overlapping information that may improve the performance.
2. **Hallucination Comparison**. The hallucination gain confirms our premise of using pre-trained LLMs help to infer or imply information when it is not provided in the context. For example, we observe a gain due to hallucination up to 11.22% for parts and 15.62% for task in the refrigerator. However, the gain from hallucination also depends on type of equipment. As mentioned, above, vending machines are less common equipment, and we only observed a gain up to 6% from hallucination.

Table 1. Results of Parts and Tasks Extraction for a Smart Refrigerator

	Part Extraction			**Task Extraction**		
Prompt Type	Precision (%)	HAL-R (%)	Precision HAL+(%	Precision (%)	HAL-R (%)	Precision HAL+ %
P-1S	77.33	6.22	83.56	71.56	15.56	87.11
P-1S+	74.81	6.11	80.92	68.32	16.41	84.73
P-1SE	**94.49**	2.94	**97.43**	84.56	**7.35**	91.91
P-1SE+	92.83	**2.51**	95.34	**84.59**	7.89	**92.47**

Table 2. Results of Parts and Task Extraction for Vending Machine

	Part Extraction			Task Extraction		
Prompt Type	Precision (%)	HAL (%)	Precision HAL+(%)	Precision (%)	HAL (%)	Precision HAL+(%)
P-1S	88.22	3.7	91.91	89.74	6.62	96.36
P-1S+	88.41	3.64	92.05	89.74	6.62	96.36
P-1SE	**95.26**	2.19	**97.45**	93.43	3.64	**97.08**
P-1SE+	94.12	**2.09**	97.21	**94.08**	**1.74**	95.82

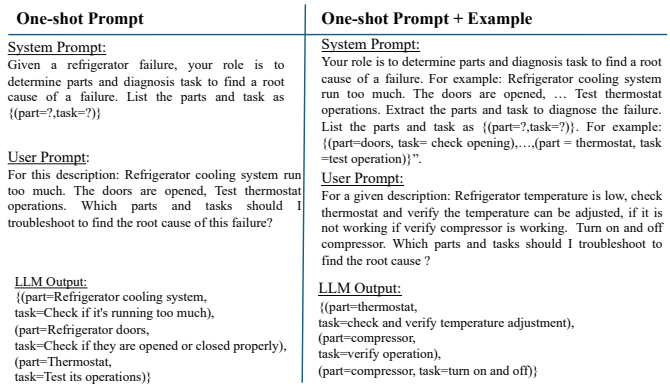

Figure 8. Examples of experiments input and outputs.

Finally, we conclude that acceptable performance is achieved by fine-tuning prompts with few instructions and relevant examples of the context. This technique minimizes the uncertainty of the equipment context in the pre-trained model. At the same, it is recommended to tailor the prompt with instructions for finding related and inferred entities, which will benefit finding correct hallucinations from common equipment.

6.2. Diagnostic State Detection Method

To evaluate the performance of the Diagnostic State *DS* detection, we designed experiments focusing on our proposed 2D area detection with nested parts. Area detection serve as a crucial component to evaluate the feasibility and effectiveness of tracking diagnostic steps and guidance indicators. Once the areas of interest and parts are identified, the system automatically use 3D positioning to determine the appropriate guidance actions. Traditional approaches have struggled to detect areas of interest across varying conditions

of equipment installment, lighting and materials Our approach also rely on small datasets while leveraging LLM.

6.2.1. Dataset settings

We collected few samples from public sources for two types of equipment: refrigerator and vending machine. We collect 200 images which were manually annotated for each type of labels: a) viewpoints e.g. front and left; b) target areas e.g. top, middle and bottom; c) parts such as control panel, water dispenser, product dispenser, etc. ,as shown in Figure 9. Each dataset was split into training, validation and testing and we also applied common data augmentation techniques to increase the dataset size. We also used private datasets, but we do not reported due to confidentiality restrictions.

6.2.2. Experiments settings

To evaluate the performance of the models, we defined two types of experiments:

1. **Equipment Areas** (Baseline): A model using only the target areas that enable tracking of technician and navigation assistant to complete the diagnostic plan. This experiment represents a traditional approach for training.
2. **Proposed Nested Areas** (Nested Areas): A model using nested object represented by parts ⊂ areas ⊂ viewpoints.

It is important to mention that a model focused solely on parts is inadequate for navigation purposes. In equipment featuring glass doors or windows, parts are commonly visible from multiple areas and viewpoints, making a part-centric model unsuitable for tracking visual observation and motion.

6.2.3. Evaluation Method

The proposed models were trained using Yolo detection networks. The Yolo's family is renowned for its real time capabilities, thus satisfying the requirement of sampling diagnostic states close to real-time for effective human interaction. We built models for object detection by training Yolo Object detector. We choose Yolo networks that are known to be low latency detectors with acceptable accuracy (>80%). We evaluated each experiment using the two versions of Yolo network: Yolov7 and Yolov9 [17,18].

Figure 9. Examples of Visual Dataset with Viewpoint+Areas of Refrigerator and Vending Machine

6.2.4. Metrics

The experiments are evaluated the traditional object detection metrics: Precision, Recall and Mean Average Precision (mAP). Precision and Recall measures the accuracy and completeness of object detection, respectively. Meanwhile, mAP combines Precision, Recall and confidence score into a single metric, which then averaged across all the classes to provide an overall measure of detection performance.

6.2.5. Results

Our goal was to boost performance in challenging areas of the equipment, specifically subject to glass or reflective areas. Therefore, we used a refrigerator and vending machine, which often features reflection and translucid areas. Table 3 and 4 provide a comparison between our proposed method and baseline approach. We can summarize the results as follows:

1. **Refrigerator Results**. The results indicate that our model effectively balances recall and precision performance, particularly in the bottom area. Specifically, in the YoloV7 column, refrigerator **front-bottom** area initially had a low recall of 56.20% and improved to 71% with our method. As seen in Figure 8, front-bottom picture is a problematic aluminum surface. Additionally, we observed the mAP50 improved 10.30% for both bottom and top areas, while the middle area remains with similar performance level. We conclude that the middle part is well defined, showing a sufficient difference compared to top and bottom areas, as illustrated in Figure 8.
2. **Vending machine Results**. The results highlights that the middle area is particularly problematic. See Figure 8 in middle area picture is a translucid area that can be confused reflections. In the YoloV7 column, the recall rate of 56.20% improved up to 75% with our method. Similarly, mAP50 shows overall improvement across all the areas.

Table 3. Results of Area Observation State of a Refrigerator

		Yolov7			Yolov9		
Exp. type	Object class	Precision (%)	Recall (%)	mAP50 (%)	Precision (%)	Recall (%)	mAP50 (%)
Baseline	Front-Top	84.60	71.00	67.10	97.50	100.00	99.30
	Front-Middle	69.20	56.20	43.80	96.70	94.50	96.20
	Front-Bottom	79.07	72.30	66.07	82.10	**100.0**	**98.80**
Nested Areas	Front-Top	86.00	69.50	70.00	**100.00**	**100.00**	**99.50**
	Front-Middle	92.10	75.00	74.40	**97.60**	**100.00**	**98.60**
	Front-Bottom	89.05	72.25	79.97	**88.20**	93.80	93.70

Table 4. Results of Area Observation State of a Vending Machine

		Yolov7			Yolov9		
Exp. type	Object class	Precision (%)	Recall (%)	mAP50 (%)	Precision (%)	Recall (%)	mAP50 (%)
Baseline	Front-Top	97.30	89.70	90.20	97.50	100.00	99.30
	Front-Middle	100.00	0.71	0.715	96.70	94.50	96.20
	Front-Bottom	100.00	56.20	56.80	82.10	100.00	98.80
Nested Areas	Front-Top	95.10	100.00	98.60	100.00	100.00	99.50
	Front-Middle	97.80	71.00	71.20	97.60	100.00	98.60
	Front-Bottom	92.30	75.00	75.40	85.90	100.00	98.40

3. **Network comparison**. We observe the YoloV7 network **benefited the most** from our method. Meanwhile, both methods exhibit excellent performance with YOLOv9, with nearly all scores above 95%, leaving little room for improvement with the nested approach. This significant improvement for small objects with YOLOv9 is attributed to its new architecture design. As reported by the authors [17], YOLOv9 introduces a new tool called Programmable Gradient Information (PGI), which adds a sidetrack to help the model retain and utilize crucial details. In our context, small objects are **often** nested within areas, which may explain why the gain was smaller with our method. **Overall**, the method shows an improvement in the complicated areas by up to 2.4 %.

Finally, we implemented a prototype as mobile App. Figure 10 shows screenshots of our system which was successfully validated with internal machinery and technicians.

7. CONCLUSION

We presented a novel framework designed to automate and facilitate the guidance for technicians performing onsite support services. Our design exploits cutting-edge technologies such as LLMs to eliminate human-intensive tasks and AR/2D detectors to enhance human interaction. The proposed methods enable a general framework which is applicable to a wide range of equipment diagnosis. To offer such as extensibility, we create a knowledge graph. We leverage ChatGPT to extract parts and task combined with image annotations to map spatial definition of the parts. Our 2D object detection model combined with 3D positioning overcomes intrinsic machinery challenges such as translucid areas and high reflection. Our experiments show the feasibility of our proposal. We implemented a prototype and conducted a system validation with real technicians. We plan to extend our approach in two dimensions: 1) detection models for diagnosis, and 2) user experience. We will investigate diagnosis recommendation methods from our sequence of images. Finally, we will explore techniques to make remote assistance more effective and immerse such as virtual reality (VR) and haptic devices.

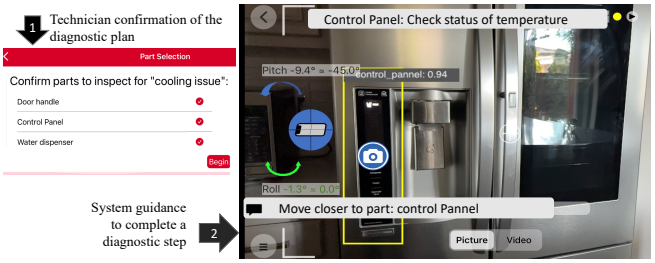

Figure 10. Screenshot of our System Prototype Implemented for Guided Diagnosis for Refrigerators.

REFERENCES

[1] Shalaby, W., Arantes, A., GonzalezDiaz, T.,; Gupta, C. (2020, June). Building chatbots from large scale domain-specific knowledge bases: Challenges and opportunities. In 2020 IEEE (ICPHM) (pp. 1-8). IEEE

[2] Gonzalez, Teresa. et. al Guided Visual Inspection enabled by AI-based Detection Models. (2021). 1-8. 10.1109/ICPHM51084.2021.9486573.

[3] Hütten, N.; Alves Gomes, M.; Hölken, F.; Andricevic, K.; Meyes, R.; Meisen, T. Deep Learning for Automated Visual Inspection in Manufacturing and Maintenance: A Survey of Open- Access Papers. *Appl. Syst. Innov.* 2024, *7*, 11.

[4] Jang, J.; Shin, M.; Lim, S.; Park, J.; Kim, J.; Paik, J. Intelligent Image-Based Railway Inspection System Using Deep Learning-Based Object Detection and Weber Contrast-Based Image Comparison. *Sensors* 2019, *19*, 4738.

[5] Cabot, P., Navigli, R. REBEL: Relation Extraction By End-to-end Language generation. In Findings EMNLP 2021

[6] Gilardi, F., Alizadeh, M., & Kubli, M. (2023). Chatgpt outperforms crowd-workers for text-annotation tasks. Proceedings of the National Academy of Sciences 2023

[7] Wang, C., Liu, X., Song, D. (2020). Language models are open knowledge graphs. arXiv preprint arXiv:2010.11967.

[8] Wadhwa S, Amir S, Wallace BC. Revisiting Relation Extraction in the era of Large Language Models. Proc Conf Assoc Comput Linguist Meet. 2023 Jul;2023:15566-15589

[9] Wang, K., Lin, Y., Weissmann, B., Savva, M., Chang, A. Ritchie, D. (2019). PlanIT: planning and instantiating indoor scenes with relation graph and spatial prior networks. ACM Transactions on Graphics. 38.

[10] Lasitha Vidyaratne1, Xian Yeow Lee1, Aman Kumar1, Tsubasa Watanabe2, Ahmed Farahat1, Chetan Gupta, ICPHM 2024

[11] Ha H, Song S. Semantic abstraction: Open-world 3d scene understanding from 2d vision-language models. In6th Annual Conference on Robot Learning 2022 Aug 15.

[12] Mendoza-Ramírez, C.E.; Tudon-Martinez, J.C.; Félix-Herrán, L.C.; Lozoya-Santos, J.d.J.; Vargas-Martínez, A. Augmented Reality: Survey. *Appl. Sci.* 2023, *13*, 10491

[13] I. Permozer and T. Orehovački, Utilizing Apple's ARKit 2.0 for Augmented Reality Application Development, (MIPRO), 2019

[14] Refrigerator Manual, URL: https://www.lg.com/us/support/manuals-documents, Visited 07/2024

[15] Vending Machine manual, https://www.royalvendors.com/customer-service/technical-info/manuals/manuals-vendors/, 07/2024

[16] Wang, Chien-Yao & Bochkovskiy, Alexey & Liao, Hong-yuan. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors.

[17] Wang, Chien-Yao et al. "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information." (2024).

[18] Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M.A., Socher, R., Amatriain, X., & Gao, J. (2024). Large Language Models.

[19] JRAutomation Product Brochure 2024, https://www.jrautomation.com/