Graph-Grounded Multimodal Agent for Trustworthy Power-Line Operations and Maintenance

Sania Verma[1,*], Joyjit Chatterjee[2,3,†], Jing Guo[4], Akash Marar[2,†], Rajendra Shankaran[5,‡], Garima Sharma[6,§], Zekun Guo[7] and Anu Mehra[8]

[1]City, St. George's University of London, London, UK
[2]EPAM Systems, London, UK
[3]Loughborough University, Loughborough, UK
[4]Takit Inc., USA
[5]Reckitt Global Technology Hub, Hyderabad, India
[6]Application Engineering Group, MathWorks Inc, Plano, Texas, USA
[7]University of Hull, Hull, UK
[8]Amity University, Noida, India

Abstract

There has been growing application of Unmanned Aerial Vehicles (UAVs) and computer vision to support safety-critical inspection activities for power transmission infrastructure. While the past work has largely focused on AI-based fault detection, trustworthy O&M decision support also requires contextualizing detected defects with component information, severity, engineering knowledge, and operational risk. Foundation Vision-Language Models (VLMs) can provide fluent and plausible-defect descriptions but remain prone to hallucination. Additionally, Retrieval-Augmented Generation (RAG) methods leveraging operations and maintenance (O&M) documentation cannot represent complex engineering dependencies in an auditable and transparent manner. We propose a graph-grounded, physics-aware agentic framework to tackle this challenge for power-line O&M, wherein domain-specific knowledge is fused from three different evidence streams: structured visual claims from a fine-tuned YOLO model combined with a VLM adapted via Many-Shot In-Context Learning, leveraging annotated defects and O&M actions from electrical engineers; a knowledge graph curated by domain specialists that integrates components, defects, severities, failure modes, engineering standards, risks, and maintenance actions; and a physics-informed simulation dataset, demonstrated through a reduced-order Stockbridge-damper case study. An agentic workflow combines these sources into a single auditable decision traceable from image to work order. Each claim is validated against the knowledge graph and physics evidence before a severity-conditioned maintenance action and work order are generated with deterministic guardrails and validation checks. On a 51-sample held out validation set, the perception pipeline achieved an F1 score of 0.83, while human evaluation of 11 work orders yielded a KG supported faithfulness score of 0.88. These results demonstrate the feasibility of evidence-grounded and traceable O&M decision support, while engineering correctness and broader operational generalisation remain subjects for further validation.

Keywords

Agentic AI, knowledge graphs, vision-language models, power-line inspection, operations and maintenance, Simulink, trustworthy AI

1. Introduction

Reliable operation of power transmission infrastructure depends on timely inspection, accurate interpretation of faults, and clearly reasoned maintenance strategies. Transmission lines face numerous

GLOW@ISWC'26: 2nd Workshop on Graph-enhanced LLMs for trustworthy Web data management, October 25–26, 2026, Bari, Italy

*Corresponding author.

†The views and methodologies presented in this paper are those of the authors and do not represent EPAM Systems.

‡The views and methodologies presented in this paper are those of the authors and do not represent Reckitt.

§The views and methodologies presented in this paper are those of the authors and do not represent MathWorks.

sania.verma@city.ac.uk (S. Verma); j.chatterjee@lboro.ac.uk (J. Chatterjee); jing@takit.ai (J. Guo); akash_marar@epam.com (A. Marar); rajendra.shankaran@reckitt.com (R. Shankaran); garimas@mathworks.com (G. Sharma); Z.Guo2@hull.ac.uk (Z. Guo); amehra@amity.edu (A. Mehra)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

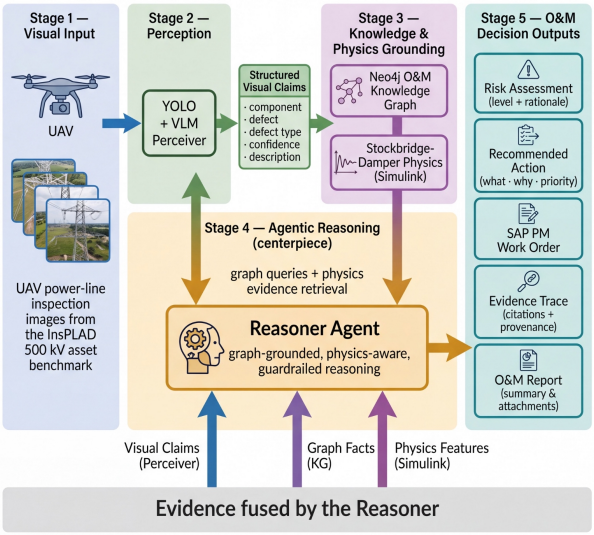


Figure 1: Overview of the proposed graph-grounded, physics-aware agentic AI framework for power-line O&M. Illustration generated with PaperBanana [5].

stressors over their lifetime that affect their performance, safety, and reliability—corrosion, mechanical loading, weather and geographic exposure, conductor vibration, contamination, and ageing—and UAVs are increasingly leveraged to inspect them, reducing the need for manual inspection and facilitating large-scale visual data collection from hazardous and complex locations [1]. Although publicly available real-world UAV imagery remains scarce, resources like InsPLAD [2] have enabled the application of Machine Learning (ML) and computer vision techniques for component detection and fault recognition.

Recent domain-specific multimodal models such as Power-LLaVA [3] demonstrate the strong potential of vision-language assistants in power-line inspection: the future of AI-based decision support in this industry rests on models that not only predict fault labels but also contextualise them with textual O&M reports, defect descriptions, and suggested maintenance strategies. Yet given the safety-critical nature of power lines, visual recognition alone is often insufficient—a VLM may detect a missing cap, a bird nest, or a damaged insulator, but practical utility of the decision depends on operational context such as the affected component, fault severity, applicable engineering and safety standards, operational risk, failure mode, and maintenance actions. Ungrounded neural language generation is prone to hallucination and therefore infeasible in these settings, and while approaches like RAG can reduce these risks by integrating language models with domain knowledge sources [4], they cannot conventionally represent engineering relationships such as defect-to-failure-mode, severity-to-action, or component-to-standard dependencies in a systematic and transparent manner.

Recent work combining Large Language Models (LLMs) with Knowledge Graphs (KGs) shows that the two play complementary roles—LLMs provide flexible natural language understanding and generation in the face of complex context, while KGs provide coherent structure, grounding, and explicit reasoning—a combination fit for power-line O&M, where maintenance recommendations must be traceable to inspection evidence, engineering rationale, and operational impact.

We therefore propose a graph-grounded, physics-aware multimodal agentic AI framework for UAV-based power-line inspection and decision support, integrating three sources of data: (1) structured visual evidence from a trained YOLO classifier (for component detection and defect isolation) coupled with a foundational VLM prompted through Many-Shot In-Context Learning on a curated corpus of defect descriptions and O&M actions annotated by electrical engineers; (2) an expert-curated, domain-specific KG of power-line components, defects, severity levels, failure modes, engineering standards, operational risks, and maintenance strategies; and (3) physics-informed simulation data for a selected component as a case study. An agentic reasoning workflow combines the extracted visual claims with KG queries and simulation evidence to generate traceable O&M recommendations, validated with a human-in-the-loop evaluation protocol. We envisage that the proposed framework can make AI-based

power-line inspection more faithful, auditable, and operationally valuable.

2. Related Work

Visual inspection and multimodal models. Research on visual power-line inspection has largely focused on turning UAV imagery into component detections, defect labels, and anomaly scores, as reviewed by Liu et al. [1] and surveyed for deep learning approaches by Faisal et al. [6], with InsPLAD providing a real-world UAV benchmark of annotated assets and defects [2]. These contributions make inspection more measurable and reproducible, but mainly evaluate perceptual outputs rather than downstream maintenance reasoning. Vision-and-language pre-training such as CLIP [7] and BLIP [8] enabled stronger linking of visual and language inputs, although reliability concerns are increasingly acknowledged [9]; in power transmission, Power-LLaVA introduced a large-scale language-and-vision inspection assistant [3]. Such assistance, however, still needs explicit grounding in asset topology, defect semantics, operational risk, and maintenance constraints before it can be trusted for O&M decisions.

Knowledge graphs for power systems and fault diagnosis. Knowledge graphs represent the entities and relationships of a domain as a queryable graph, allowing for semantic representation, validation, and provenance [10]. In the electricity domain, GridKG represents distribution-grid data as a graph [11], later extended with topology, asset metadata, maintenance information, and outage events for decision support [12]; KGs have also been applied to power-system fault diagnosis [13] and modelling decarbonisation [14]. Related maintenance domains show similar benefits, including KG-driven fault diagnosis in intelligent manufacturing [15] and fault KGs for aircraft prognostics and health management [16], alongside growing interest in graph learning for grid problems such as graph reinforcement learning [17]. Nonetheless, multimodal UAV inspections validating VLM claims against graph-engineered knowledge remain unexplored.

Grounding language agents with structured and multimodal knowledge. Retrieval-augmented generation reduces parameter dependence by conditioning on retrieved evidence [4], but retrieved passages are not an auditable engineering model of components, defects, severity levels, and actions. A growing number of studies combine KGs with LLMs, from surveys of LLM–KG question answering [18] to graph-based retrieval such as GraphRAG [19]. Work on multimodal KGs—the large-scale entity–image links of MMpedia [20], reasoning with graph-structured cross-modal knowledge [21], and explainable language reasoning with multimodal KGs [22]—shows how knowledge structure improves cross-modal inference, while digital-twin research has scaled KGs to help automate AI for operational assets [23]. For power-line O&M, however, these strands have not yet been combined to convert VLM-derived observations into traceable claims that are validated against KGs and enriched with physics-informed evidence.

Recent research has shown advancements in multimodal and agentic grounding – such as multimodal KG–RAG for Visual Question Answering (VQA) [24, 25] and typed-KG grounding for industrial asset operations [26, 27, 28], but to the best of our knowledge, no past literature has focused on validating visual claims against a domain-specific power-line O&M graph while also considering physics-informed simulation knowledge under deterministic guardrails, as we showcase in this paper.

Trustworthy, explainable, and agentic decision support. Because safety is critical for power-line O&M, recommendations need to be explainable and auditable; explainable AI for energy-systems maintenance has been reviewed in [29]. Agentic and multi-agent designs increasingly structure such reasoning: AGENTiGraph lets LLM-based agents interact with and update domain-specific KGs through natural language [30], a collaborative AI–human multi-agent system with a KG has been proposed for trustworthy diagnosis prediction [31], and semantically enriched KGs have driven multi-agent inspection planning for non-destructive testing [32]. Agentic AI frameworks for smart grids have particularly emphasised on autonomy, safety, and explainability [33]. Collectively, these works support

Table 1
Composition of the ingested InsPLAD data within the Power-Line O&M Knowledge Graph.

Graph content	Count
`Image` — detection subset	10,561
`Image` — defect-supervised subset	11,525
`Image` — anomaly-detection subset	27,620
`Image` **total**	**49,706**
`ImageAnnotation` (COCO bounding boxes)	28,959
`Curated VLM Observation records`	255
`SimulationRun` (Stockbridge damper)	80

our design choice of a graph-grounded, human-in-the-loop agentic workflow in which each maintenance recommendation is traceable to inspection evidence, engineering rationale, and operational impact.

3. Dataset Description and Pre-Processing

The dataset is taken from the InsPLAD power-line asset inspection benchmark [2]. It focuses on a 500 kV transmission asset (STN, northeast Brazil) and comprises object detection (10,561 images), defect-supervised classification (11,525 images), and unsupervised anomaly detection (27,620 images), with 28,959 COCO-format image annotation records. All three subsets were used to construct the Power-Line O&M Knowledge Graph (Section 5), providing broad component and asset coverage.

A subset of 255 images was separately curated with multi-field annotations for: *component type, defect presence (True/False), defect description, and confidence score.* This subset supported supervised fine-tuning of the pipeline. The expert-curated corpus is limited in scale, but remains consistent with recent civil-infrastructure VLM work on 254 expert-scored bridge damage images [34], reflecting a broader trend towards small, high-integrity corpora in inspection domains. The corpus covers 15 components and six defect-type classes (five defects plus a non-defective "good" condition). Of these images, 85 were annotated by human domain experts as ground truth. The remaining 170 images were annotated using GPT-5.5-pro.[1]

3.1. Data Pre-Processing

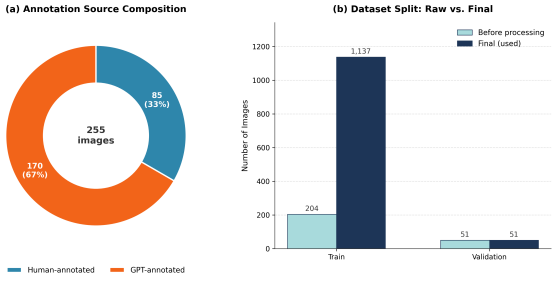
Since directly fine-tuning a VLM on this limited corpus led to overfitting (elaborated in Section 4.2), a YOLO classifier was fine-tuned instead for component classification and defect detection. The 255 images were organized into 27 component-defect classes across the 15 components.[2] The dataset was split 80:20 into training and validation sets, respectively. Different augmentation techniques were applied to the training set (*flipping, brightness/contrast, rotation, affine transforms, and Gaussian blur/noise*). These techniques were chosen to preserve the geometric and structural cues. Figure 2a summarizes the resulting dataset composition.

3.2. Human Evaluation

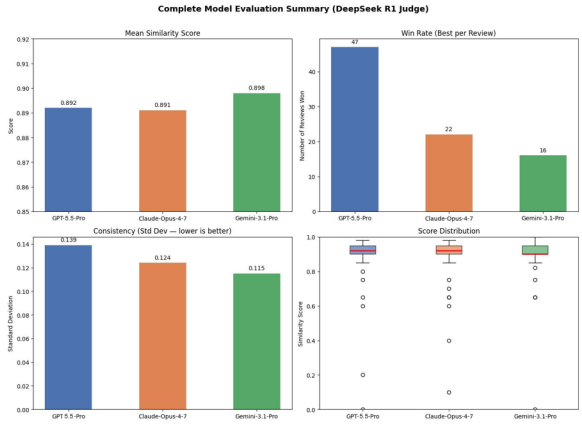
We extended the InsPLAD dataset with fault descriptions, impact assessments, and repair instructions. The annotators were power-line inspection specialists with certifications in high-voltage power-line maintenance. The annotators were selected via a test process in which they were asked to identify fault classes and component names independently. The annotator with the highest accuracy on the test set was selected to complete the entire data corpus.

[1] Selected after comparing annotation quality across GPT-5.5-pro, Claude-Opus-4-7, and Gemini-3.1-pro-preview, judged against the human ground truth by DeepSeek Reasoner as an independent LLM judge (Figure 2b); GPT-5.5-pro was closest to the human annotations in 47 of 85 cases.

[2] For three components—spacer, spiral damper, and yoke—no defect-present examples were available in the curated dataset, so only the defect-absent case was considered for these components.



(a) Dataset composition and preprocessing summary: annotation source split (85 human, 170 GPT-annotated) and raw vs. final image counts for training/validation sets.



(b) LLM annotation agreement with human ground truth, judged independently by the DeepSeek LLM judges across mean similarity, win rate, consistency, and score distribution.

Figure 2: (a) Dataset preprocessing overview. (b) LLM annotator evaluation against human-expert ground truth.

4. Methodology

4.1. Framework Overview

Figure 1 illustrates the overall pipeline of the proposed work, organized into five layers.

1. *Evidence Input Layer*: UAV-captured power-line inspection images obtained from the InsPLAD dataset.
2. *Perception Layer*: Turns each image into a structured visual claim: a fine-tuned YOLO classifier outputs the component, defect presence, and a confidence score, which the VLM Perceiver Agent (a vision-language model guided by many-shot in-context learning) enriches with the affected part, the nature and degree of the defect, a confidence value, and visual proof (detailed in Section 4.2). The resulting claim is verified in the subsequent layers.
3. *Knowledge and Physics Grounding Layer*: Checks each visual claim against the Power-Line O&M Knowledge Graph (Section 5)—properties, failure modes, severities, standards, and risks—and, for the Stockbridge damper case study, against physics-informed simulation evidence (Section 5.1) as an independent, non-visual signal.
4. *Agentic Reasoning Layer*: The Reasoner Agent (Section 5.2) fuses the visual claim, graph facts, and simulation attributes into a decision: a deterministic guardrail holds severity to the more conservative of the visual and physics readings, and the knowledge graph then determines which maintenance action that severity requires.
5. *O&M Decision Outputs Layer*: Returns a risk assessment, a recommended action, a pre-filled work order, and an end-to-end evidence trace as the output O&M report.

The workflow follows an *input* $\rightarrow$ *agent* $\rightarrow$ *grounding* $\rightarrow$ *validation* $\rightarrow$ *output* approach. Visual evidence is obtained through YOLO and the VLM while the Reasoner Agent integrates it along with KG and Physics-based evidence. The deterministic guardrails validate severity, actions, numerical consistency, and grounding to finally output a traceable O&M recommendation and work order.

4.2. Visual Evidence Extraction

The visual evidence extraction pipeline is designed in two stages: a fine-tuned YOLO classifier, followed by many-shot VLM inference. Figure 3 shows the perception-layer workflow transforming a UAV image into a structured visual claim. This design was chosen for the following reasons.

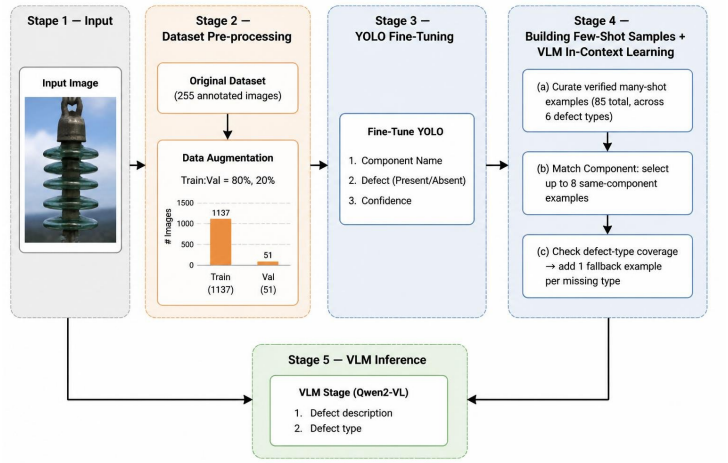


Figure 3: End-to-end workflow for visual evidence extraction. Pipeline parameters are summarized in Algorithm 1. The 255-image corpus was split into 80:20 (204 train/val), with augmentation (flipping, brightness/contrast, rotation, affine transforms and Gaussian blur/noise) applied to the training set, obtaining 1137 images.

1. **Limited annotated data**: Annotation demands significant time and domain-expert effort, making end-to-end VLM fine-tuning impractical. An early attempt to fine-tune a VLM (Qwen2-VL-2B-Instruct) on the small corpus led, as expected, to overfitting, with the model wasting token budget on memorizing longer descriptions rather than learning generalizable defect features. This motivated the two-stage design: a compact YOLO classifier fine-tuned for component classification and defect detection, with the VLM kept frozen and guided by many-shot in-context learning rather than parameter updates.
2. **YOLO's detection strength**: YOLO is state-of-the-art for component-level object detection and classification, making it well suited to identifying the component and the presence of a defect before language reasoning is applied.
3. **Deployment and iteration constraints**: Qwen2-VL-2B-Instruct was selected over the other large proprietary VLMs due to the latency [35] and computational cost constraints of UAV/edge deployment and iterative development.
4. **Reducing VLM overload and false positives**: Provision of YOLO outputs before VLM inference reduces the number of attributes the VLM must infer directly from the image which improves reliability and reduce false positives.

Detection: YOLOv11 (nano model YOLOv11n-cls) was fine-tuned to output the component type, defect presence, and a confidence score for each image.
Curated many-shot corpus and runtime selection: 85 verified-correct examples were curated from the training data across the six classes (five defect types plus the non-defective "good" condition). At runtime, up to eight same-component samples are selected from this corpus to prevent hallucination and reduce noise in VLM learning; if any class remains unrepresented, one example of each missing class is added so that every class is demonstrated in-context.
VLM inference: The VLM generates the "Defect type" and "Description". The combined YOLO+VLM pipeline produces a structured visual claim containing the `component`, `defect status`, `type`, `confidence` and `description`, which is altogether passed into the knowledge graph. The pipeline uses 25 samples per class, up to 15 filtered examples per class (85 total), and 8 same-component examples per query, with one fallback per missing class; examples are filtered by YOLO component and defect-flag agreement upon comparing with the ground truth, followed by which Qwen2-VL-2b-Instruct (4-bit NF4) processes images up to 320px.

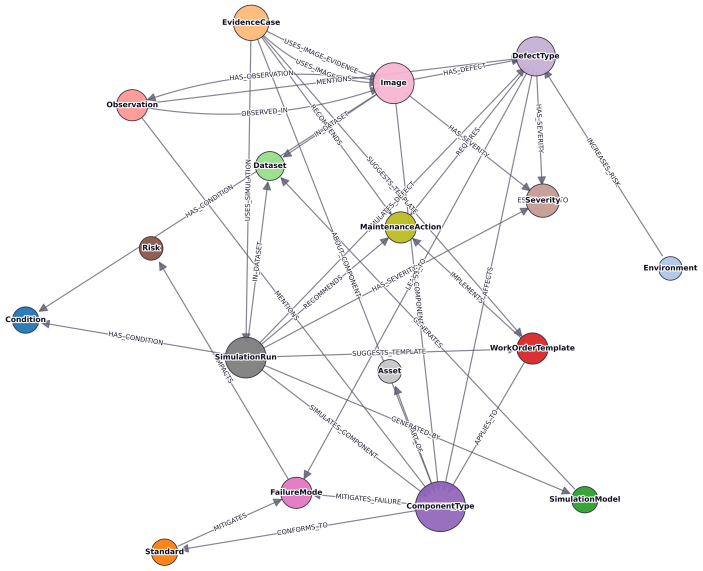


Figure 4: Overview of the ontology schema.

5. Methodology for Constructing the O&M Knowledge Graph

We construct the O&M knowledge graph as an ontology-grounded, multi-layered representation linking visual inspection footage to actionable maintenance reasoning. Following a data-driven, bottom-up process that mimics real-world power-line maintenance practice, an ontology derived from real world inspection data is progressively enriched with expert reasoning annotations, VLM text annotations, physics based simulations, explicit failure-mode, operational-impact, risk, and maintenance actions.[3]

The central concept is the physical line element – *ComponentType* class (e.g., *stockbridge_damper*, *glass_insulator*, *yoke*). Around this, and in the manner of grouping descriptive node classes adopted in [12], we organize the remaining concepts into seven interdependent layers: (i) an *Asset Hierarchy* layer for transmission-line elements and their assembly hierarchy; (ii) an *Image & Annotation Processing* evidence layer that captures UAV Images and their generated labels and descriptions; (iii) a *Defect, Severity & Escalation* layer for perceived visual defects, severity, escalation and impacts; (iv) an *Environmental & Standards Context* layer for environmental factors and relevant standards; (v) an *Evidence & Simulation* layer storing physics evidence from *SimulationModel*, *SimulationRun*, *EvidenceCase* for component and defect impact analysis; (vi) a *Machine Learning* layer that stores image and component embeddings; (vii) a *Maintenance & Work Orders* layer capturing *MaintenanceAction*, *WorkOrderTemplate* for recommended maintenance actions. The schema is implemented in Neo4j and encoded in Cypher. Our KG unifies visual, physical, environmental, regulatory, and operational knowledge to provide a rich, semantically structured context for AI reasoning agents. The principal relations are summarized in Table 2. Ontology complianceis enforced: an automated validator checks structuralinvariants over the graph,exactly one severity-conditioned REQUIRES edge per(defect, severity) pair via the when_severity property, no defect linked to severity level 0, a complete ESCALATES_TO chain 0->1->2->3, and consistency between the decision rules, the ingestion manifest, and simulation-derived actions and all checks passed on the released graph. The validation and repaid scripts will be released alongside the KG.

[3]The data used in this KG is from the InsPLAD dataset [2]. The ontology encodes 18 types of components from primary components and support hardware, 5 types of visual defects, a four-level ordinal severity scale carrying risk weights, 8 failure modes, 6 operational risks, 3 environmental contexts, 7 international and national standards, 8 maintenance actions, and 3 work-order templates.

Table 2
Principal relations of the O&M knowledge-graph vocabulary.

Relation	Domain → Range	Semantics
Asset Hierarchy		
PART_OF / ATTACHES_TO	ComponentType → Asset / ComponentType	Asset membership; assembly and attachment
Image & Annotation Processing		
HAS_COMPONENT / HAS_DEFECT / HAS_CONDITION	Image → ComponentType / DefectType / Condition	Component, visual defect, and normal-vs-defective condition of an image
DERIVED_FROM	Image → Image	Provenance: augmented to raw image
ANNOTATES / DESCRIBES	ImageAnnotation / VLMObservation → ComponentType	Ground-truth label; VLM-generated description
Defect, Severity & Escalation		
HAS_SEVERITY / ESCALATES_TO	HAS_SEVERITY: DefectType → Severity; ESCALATES_TO: Severity → Severity	Severity grading (0–3); ordinal chain $0 \to 1 \to 2 \to 3$
LEADS_TO / IMPACTS	LEADS_TO: Severity → FailureMode; IMPACTS: FailureMode → Risk	Failure mode induced by a severity level; risk exposure from a failure
AFFECTS	DefectType → ComponentType	Components a defect can affect
Environmental & Standards Context		
INCREASES_RISK	Environment → DefectType	Environmental factors amplifying a defect
CONFORMS_TO / MITIGATES	CONFORMS_TO: ComponentType → Standard; MITIGATES: Standard → FailureMode	Compliance standards; standard controlling a failure mode
Evidence & Simulation		
USES_IMAGE / USES_SIMULATION	EvidenceCase → Image / SimulationRun	Multimodal fusion into one case
GENERATED_BY / SIMULATES_COMPONENT	SimulationRun → SimulationModel / ComponentType	Physics model producing the run; component under simulation
Machine Learning		
EMBEDS / MODELS_COMPONENT	EmbeddingRun → Image / ComponentType	Image crops embedded in latent space; component type in embedding
Maintenance & Work Orders		
REQUIRES	DefectType → MaintenanceAction	Action required (conditioned via `when_severity`)
SUGGESTS_TEMPLATE	EvidenceCase → WorkOrderTemplate	Decision case to recommended template
IMPLEMENTS / APPLIES_TO	WorkOrderTemplate → MaintenanceAction / ComponentType	Template realizing an action; target component

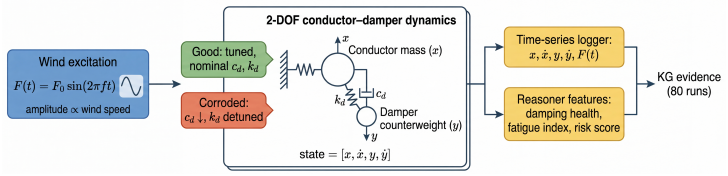


Figure 5: Reduced-order Simulink evidence generator for the Stockbridge damper case study. The model was run 80 separate times, each with a different parameter combination – varying the excitation frequency, wind-speed-driven force amplitude, and the corrosion factor (good vs corroded scenarios)

5.1. Stockbridge Damper Physics Evidence

The physics-evidence stream is deliberately scoped to a single component, the *stockbridge_damper*, which we later demonstrate as a case study. Its role in the framework is to generate structured, physics-informed evidence for graph-grounded O&M reasoning, rather than to act as a certified mechanical design simulator or as a simulator for all power-line components. We model the Stockbridge damper as a reduced-order two-degree-of-freedom conductor-damper vibration system, where x denotes the local conductor displacement near the attachment point and y denotes the effective damper counterweight or messenger displacement. Wind-induced aeolian excitation is represented by a sinusoidal forcing input $F(t) = F_0 \sin(2\pi f t)$, with the force amplitude parameterized by wind speed and the excitation frequency sampled across the simulation runs.

The healthy damper is treated as a tuned mass damper attached to the first local conductor vibration mode. Corrosion is encoded as two coupled physical effects: loss of damper damping and detuning of damper stiffness. In the generated scenarios, good-condition dampers use low corrosion factors, while corroded dampers use higher corrosion factors that reduce energy dissipation and shift the damper away from the tuned condition. The model therefore provides qualitative evidence about how visible corrosion can translate into poorer vibration control and increased fatigue-related risk.

For each simulation run, Simulink logs time-series signals,[4] that are summarized into reasoner-facing features: conductor RMS and peak displacement, relative RMS displacement, fatigue damage index, damping health index, and risk score. These runs are exported as JSONL records linking the component, simulated condition, severity, VLM image example, features, and recommended O&M action. In the KG, *SimulationRun* and *EvidenceCase* nodes connect visual Stockbridge damper evidence with physics features and maintenance actions, enabling the agent to justify recommendations through a traceable path combining image observations, simulation evidence, and ontology-grounded defect-to-action knowledge.

5.2. Agentic Reasoning for O&M Recommendation Generation

The ***Reasoner Agent*** is a two-turn, graph-grounded LLM agent that maps multimodal input, which includes VLM observations, Neo4j O&M knowledge, and Simulink physics information, into explainable O&M decisions. VLM predictions are re-grounded through a comparison of their severity against that of corresponding physics predictions, with the more conservative assessment that determines the final severity. The Knowledge Graph maps the severity into the maintenance operation, while deterministic guardrails provide for safety and auditability. The agent goes through five steps, namely, *Retrieve*, *Bound*, *Reason*, *Validate*, and *Author*.

1. **Retrieve.** The agent assembles a structured EvidencePack of (i) VLM outputs (detected components, defects, severity, and confidence); (ii) Simulink physics results; and (iii) a multi-hop knowledge-graph context spanning component properties, failure modes, risk propagation, maintenance actions, asset relationships, environmental factors, and neighboring defects.
2. **Bound.** A deterministic guardrail computes a severity floor on a three-level ordinal scale (minor, moderate, severe): the maximum of the VLM-assigned level and the level banded from the matched simulation run's risk score (below 0.35 minor, 0.35–0.60 moderate, 0.60 and above severe), escalated one further level if that run's damping health index falls below 0.15. The KG encodes a four-level ordinal scale 0–3, where level 0 denotes normal condition and is never linked from a *DefectType*; the reasoner therefore operates on the three defect-present levels 1–3, referred to as minor, moderate, and severe. The knowledge graph is then queried for the action its REQUIRES relation attaches to the resulting level, so that corrosion resolves to visual monitoring, hardware replacement, or emergency intervention according to severity rather than one fixed response per defect type.
   floor = max(sev_VLM, band(risk_score)), escalated one level if damping_health < 0.15 and capped at severe; the LLM may adopt a higher severity than the floor but the Validate stage clamps any lower one.
3. **Reason.** The LLM generates a structured maintenance decision: severity assessment, risk evaluation, failure modes, recommended maintenance action, supporting evidence citations, and a reasoning summary. The prompt prioritizes physics and knowledge-graph evidence over visual observations when they contradict, reflecting their higher reliability relative to VLM predictions
4. **Validate.** The output is checked against guardrails verifying severity constraints, ontology-compliant maintenance actions, numerical consistency, and citation correctness. Violations trigger one automatic re-prompt, and persistent violations are corrected through deterministic rule-based clamping, so the final decision always satisfies safety requirements.
5. **Author.** To separate reasoning from presentation, a second LLM invocation generates a SAP Plant Maintenance (PM)-style work order together with a concise O&M report.

[4] Conductor displacement, conductor velocity, damper displacement, damper velocity, relative displacement, dynamic strain proxy, fatigue damage rate, and wind force.

6. Experiments

6.1. Experimental Design

Perception Pipeline Configuration: YOLOv11, being state-of-the-art (see Section 4.2), was fine-tuned on the 27-class structure described in Section 3.1. The model was fine-tuned for 30 epochs. Given an image, YOLO produced the following output: `[Component_name, Defect (Present/Absent), Confidence Score]`

Many-Shot Example Curation: Examples were curated in two stages (Algorithm 1). First, up to $n = 25$ samples per defect-type class (six classes) were verified against the YOLO predictions and capped at $k = 15$ per class, yielding 85 curated samples. At runtime, up to $m = 8$ samples are selected per query, as depicted in Figure 3.

Algorithm 1 Many-Shot Example Curation and Runtime Selection

Require: training set D; classes $C = \{\text{good, rust, torn-up, missingcap, peeling-paint, nest}\}$; pool size n=25; per-class cap k=15; same-component cap m=8

Phase A — Offline curation (build E_{full} once): $E_{full} \leftarrow \emptyset$

for each $c \in C$ **do**
  $S_c \leftarrow$ random sample of $\min(n, |D_c|)$ images from D with defect_type $= c$
  $V_c \leftarrow \{I \in S_c : \text{YOLO}(I).\text{comp} = \text{component}(I) \wedge \text{YOLO}(I).\text{def} = \text{defect_present}(I)\}$
  $E_{full} \leftarrow E_{full} \cup$ first $\min(k, |V_c|)$ elements of V_c
end for

Phase B — Runtime selection (per query I_q, component comp_q):

$E_q \leftarrow$ first m examples from E_{full} with component $= \text{comp}_q$
for each $c \in C$ with no example in E_q **do**
  add one example of class c from E_{full} to E_q ▷ fallback
end for
return E_q

6.1.1. VLM and Reasoner Configuration

The VLM stage uses Qwen2-VL-2B-Instruct (4-bit NF4, max 320 px) to produce the defect description and defect type per query. The Reasoner Agent (Section 5.2) is implemented using `openai/gpt-oss-120b`,[5] operating over the five-stage pipeline described in Section 5.2.

6.1.2. Knowledge Graph and Simulation Evidence

To support agentic reasoning, we prepared a multimodal evidence corpus integrating visual perception with physics-informed simulation. The full InsPLAD image inventory is ingested into the graph as 49,706 image records, of which 10,561 carry object-detection annotations (Table 1). From this inventory, we curate an evaluation subset of 255 VLM evidence records, each pairing a component label, defect label, severity, and a natural-language observation produced by the VLM. In parallel, 80 Stockbridge-damper runs from the reduced-order Simulink model are stored as *SimulationRun* nodes with physics-informed features (vibration amplitude, damping health, fatigue proxy, and risk score) and linked to the corresponding image examples through *EvidenceCase* nodes, yielding structured cases that combine visual evidence, simulation evidence, and ontology-grounded maintenance recommendations (Section 5.1).

7. Results and Discussion

7.1. Perception and Reasoning Performance

In this section, we evaluate the two stages of perception layer, YOLO and the VLM, on the validation set (51 samples), as shown in Figure 6. The fine-tuned YOLO model achieved a precision of 0.86, recall

[5] Served via the Groq inference API.

Table 3
Representative pipeline examples, with predicted component, defect, and type matching ground truth.

Image	YOLO	VLM	Ground Truth
	lightning_rod_susp.; defect present; conf. 0.86	corrosion: rust visible on the suspension arm, which can weaken the structure	lightning_rod_susp.; corrosion: rust clearly visible on the suspension hardware
	polymer_insulator; defect present; conf. 0.999	torn_up: upper shed torn and deformed with a lifted skirt, reducing insulation surface and trapping contamination	polymer_insulator; torn_up: damage on the insulator sheath with a torn shed

of 0.79, and F1 score of 0.83 on component classification and defect presence/absence detection. The VLM was prompted via many-shot in-context learning. This yielded a defect-type accuracy of 0.75 and a description similarity of 0.65 (ground-truth and predicted descriptions were compared using cosine similarity). The gap in accuracy between YOLO's precision/recall and the VLM's output is expected: YOLO is fine-tuned end-to-end on the full 27-class structure, whereas the VLM must differentiate among six defect types at inference time (Figure 3) without any parameter updates. Table 3 presents two representative evidence cases showcasing the outputs of the pipeline.

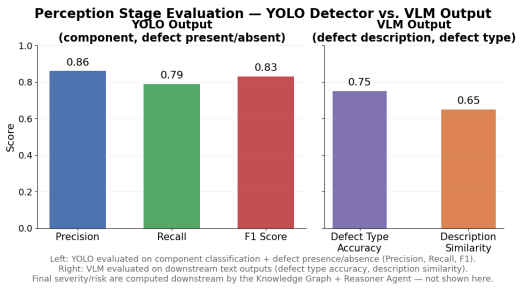


Figure 6: Perception stage evaluation on the held-out validation split: YOLO precision/recall/F1 (left) and VLM defect type accuracy/description similarity, computed via semantic similarity (right).

7.2. Faithfulness and Human Evaluation

A human expert evaluated faithfulness, defined as the fraction of claims in the agent's work-order output supported by the retrieved knowledge-graph (KG) context. *Faithfulness measures alignment with the KG support; it does not assess correctness against the physical asset, image, engineering standards, or the optimal maintenance action.* Across 11 work orders, the faithfulness score was 0.88, demonstrating the effectiveness of KG grounding in reducing unsupported claims.

7.3. Case Study

Figure 7 illustrates a sample scenario with a **corroded Stockbridge damper** on the 500 kV STN asset. According to the VLM Perceiver, the defect is level 2 out of 3 (moderate) corrosion (confidence 1.00). This severity claim is corroborated by evidence from the physics simulation (damping health index = 0.347, risk score = 0.479) and the knowledge graph, in which corrosion is related to mechanical failure and conductor-drop risk (0.95) and, at this severity level, requires hardware replacement according to IEC 61284. The Reasoner Agent upholds this severity, adopts the more cautious risk assessment from the KG over the simulation-derived one, satisfies all validation guardrails, and outputs the work order.

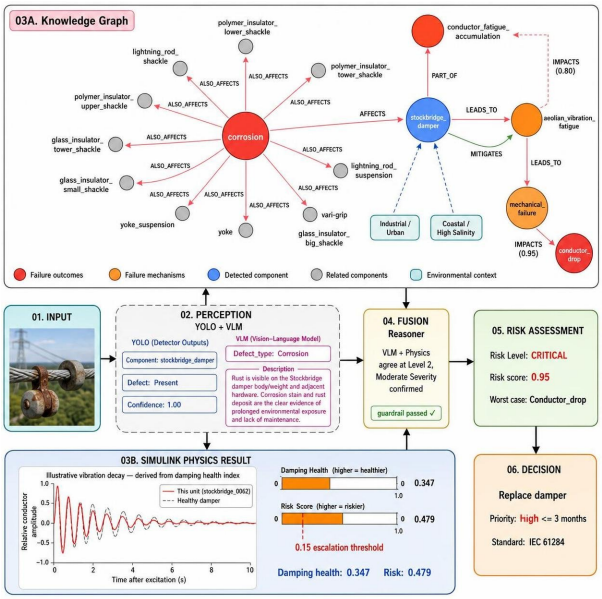


Figure 7: End-to-end case study of the UAV power line inspection pipeline, illustrating the progression from defect detection to risk assessment and the resulting maintenance decision.

7.4. Limitations and Future Work

The key limitation is the small human-annotated data set used for in-context learning of the VLM, consisting of only 255 annotated images, which may restrict the model's ability to generalize to unseen defect characteristics, imaging conditions, and rarer classes. This also precluded fine-tuning the VLM. Since the framework was developed and evaluated using data from a specific operational environment, its performance in other industrial settings still requires validation. Moreover, structural ontology validation ensures conformity to the predefined schema but does not guarantee the semantic correctness of retrieved facts or image-to-graph mappings. Errors introduced at these stages may propagate through downstream components and ultimately affect the reliability of the system outputs. In future work, we will expand the dataset to fine-tune and release a domain-specific open-source VLM covering a broader range of cases and conditions. We will also pursue further verification and validation, whilst maintaining system-level testing in a simulated environment. Accordingly, the reported faithfulness score should be interpreted as evidence of grounding consistency rather than validation of real-world engineering correctness.

8. Conclusion

This work proposes an agentic framework for power-line O&M based on structured visual claims from a two-stage perception pipeline, an expert-curated KG, and a Reasoner Agent that evaluates VLM-generated claims against both graph and physical evidence. Even with limited amounts of labeled data, which is common in industrial applications, our perception pipeline managed to attain an F1 score of 0.83 in component/defect classification and 0.75 in defect type recognition. The end-to-end case study also showcases the possibility of establishing a traceable evidence chain to the work orders generated, showing the promise of context engineering in GenAI for power-line O&M. The evaluation is limited by 51-validation set of images and 11 work orders that were generated as the output of the framework, which embeds robustness.

Acknowledgments

We sincerely thank The MathWorks, Inc. for their support and the advanced capabilities provided by Simulink® software. The content of this article is owned by the authors, not by The MathWorks, Inc.

9. Declaration on Generative AI

ChatGPT, Claude and Gemini were used for LLM annotations of power line images, while DeepSeek was used as an independent judge in this paper. PaperBanana was used to generate the illustrations featured in the paper (Figures 2 and 7). The authors have reviewed and edited the generated material accordingly and fully assume responsibility for the content of the paper.

References

[1] X. Liu, X. Miao, H. Jiang, J. Chen, Data analysis in visual power line inspection: An in-depth review of deep learning for component detection and fault diagnosis, Annual Reviews in Control 50 (2020) 253–277. doi:10.1016/j.arcontrol.2020.09.002.

[2] A. L. B. Vieira e Silva, H. de Castro Felix, F. P. M. Simões, V. Teichrieb, M. M. dos Santos, H. Santiago, V. Sgotti, H. B. Lott Neto, Insplad: A dataset and benchmark for power line asset inspection in uav images, International Journal of Remote Sensing 44 (2023) 7294–7320. doi:10.1080/01431161.2023.2283900.

[3] J. Wang, M. Li, H. Luo, J. Zhu, A. Yang, M. Rong, X. Wang, Power-llava: Large language and vision assistant for power transmission line inspection, in: 2024 IEEE International Conference on Image Processing (ICIP), IEEE, 2024, pp. 963–969. doi:10.1109/ICIP51287.2024.10648271.

[4] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive nlp tasks, in: Advances in Neural Information Processing Systems (NeurIPS), volume 33, Curran Associates, Inc., 2020, pp. 9459–9474.

[5] D. Zhu, R. Meng, Y. Song, X. Wei, S. Li, T. Pfister, J. Yoon, Paperbanana: Automating academic illustration for ai scientists, arXiv preprint arXiv:2601.23265 (2026).

[6] M. A. A. Faisal, I. Mecheter, Y. Qiblawey, J. Hernandez Fernandez, M. E. H. Chowdhury, S. Kiranyaz, Deep learning in automated power line inspection: A review, Applied Energy 385 (2025) 125507. doi:10.1016/j.apenergy.2025.125507.

[7] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, in: Proceedings of the 38th International Conference on Machine Learning (ICML), volume 139 of *Proceedings of Machine Learning Research*, PMLR, 2021, pp. 8748–8763.

[8] J. Li, D. Li, C. Xiong, S. C. H. Hoi, BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation, in: Proceedings of the 39th International Conference on Machine Learning (ICML), volume 162 of *Proceedings of Machine Learning Research*, PMLR, 2022, pp. 12888–12900.

[9] K. Vu, P. Lai, Trustworthiness in vision-language models, in: Computational Data and Social Networks (CSoNet 2024), volume 15417 of *Lecture Notes in Computer Science*, Springer, 2025, pp. 73–85. doi:10.1007/978-981-96-6389-7_7.

[10] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, S. Kirrane, J. E. Labra Gayo, R. Navigli, S. Neumaier, A.-C. Ngonga Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann, Knowledge graphs, ACM Computing Surveys 54 (2021) 1–37. doi:10.1145/3447772.

[11] Y. Kor, L. Tan, M. Z. Reformat, P. Musilek, GridKG: Knowledge graph representation of distribution grid data, in: 2020 IEEE Electric Power & Energy Conference (EPEC), IEEE, 2020, pp. 1–5. doi:10.1109/EPEC50724.2020.9300437.

[12] Y. Kor, L. Tan, P. Musilek, M. Z. Reformat, Integrating knowledge graphs into distribution grid decision support systems, Future Internet 16 (2024) 2. doi:10.3390/fi16010002.

[13] C. Li, B. Wang, A knowledge graph method towards power system fault diagnosis and classification, Electronics 12 (2023) 4808. doi:10.3390/electronics12234808.

[14] W. Xie, F. Farazi, J. Atherton, J. Bai, S. Mosbach, J. Akroyd, M. Kraft, Dynamic knowledge graph

approach for modelling the decarbonisation of power systems, Energy and AI 17 (2024) 100359. doi:10.1016/j.egyai.2024.100359.

[15] C. Cai, Z. Jiang, H. Wu, J. Wang, J. Liu, L. Song, Research on knowledge graph-driven equipment fault diagnosis method for intelligent manufacturing, The International Journal of Advanced Manufacturing Technology 130 (2024) 4649–4662. doi:10.1007/s00170-024-12998-x.

[16] X. Meng, B. Jing, S. Wang, J. Pan, Y. Huang, X. Jiao, Fault knowledge graph construction and platform development for aircraft PHM, Sensors 24 (2024) 231. doi:10.3390/s24010231.

[17] M. Hassouna, C. Holzhüter, P. Lytaev, J. Thomas, B. Sick, C. Scholz, Graph reinforcement learning for power grids: A comprehensive survey, Energy and AI 23 (2026) 100671. doi:10.1016/j.egyai.2025.100671.

[18] C. Ma, Y. Chen, T. Wu, A. Khan, H. Wang, Large language models meet knowledge graphs for question answering: Synthesis and opportunities, in: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP 2025), 2025, pp. 24578–24597.

[19] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, J. Larson, From local to global: A graph RAG approach to query-focused summarization, arXiv preprint arXiv:2404.16130 (2024). doi:10.48550/arXiv.2404.16130.

[20] Y. Wu, X. Wu, J. Li, Y. Zhang, H. Wang, W. Du, Z. He, J. Liu, T. Ruan, MMpedia: A large-scale multi-modal knowledge graph, in: The Semantic Web – ISWC 2023, volume 14266 of *Lecture Notes in Computer Science*, Springer, 2023, pp. 18–37. doi:10.1007/978-3-031-47243-5_2.

[21] J. Lee, Y. Wang, J. Li, M. Zhang, Multimodal reasoning with multimodal knowledge graph, arXiv preprint arXiv:2406.02030 (2024). doi:10.48550/arXiv.2406.02030.

[22] C. Lu, J. Luo, K. Yu, T. Chen, X. Wang, F. Qian, C. Xi, Towards explainable language reasoning via multi-modal knowledge graphs, Scientific Reports 16 (2026). doi:10.1038/s41598-026-51127-4.

[23] J. Ploennigs, K. Semertzidis, F. Lorenzi, N. Mihindukulasooriya, Scaling knowledge graphs for automating AI of digital twins, in: The Semantic Web – ISWC 2022, volume 13489 of *Lecture Notes in Computer Science*, Springer, 2022, pp. 810–826. doi:10.1007/978-3-031-19433-7_46.

[24] X. Yuan, L. Ning, Q. Ye, W. Fan, Q. Li, mkg-rag: Leveraging multimodal knowledge graphs in retrieval-augmented generation for knowledge-intensive vqa, 2026. URL: https://arxiv.org/abs/2508.05318. doi:https://doi.org/10.1145/3805712.3809680. arXiv:2508.05318.

[25] X. Wan, H. Yu, Mmgraphrag: Bridging vision and language with interpretable multimodal knowledge graphs, 2026. URL: https://arxiv.org/abs/2507.20804. arXiv:2507.20804.

[26] M. Mandarapu, S. Kunkunuru, Knowledge graphs as the missing data layer for llm-based industrial asset operations, 2026. URL: https://arxiv.org/abs/2605.26874. arXiv:2605.26874.

[27] F. O'Donncha, N. Zhou, N. Martinez, J. T. Rayfield, F. F. Heath III, A. Langbridge, R. Vaculin, Evidence-driven reasoning for industrial maintenance using heterogeneous data, in: Y. Li, G. Rehm, M. Tu (Eds.), Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track), Association for Computational Linguistics, San Diego, California, USA, 2026, pp. 736–748. URL: https://aclanthology.org/2026.acl-industry.50/. doi:10.18653/v1/2026.acl-industry.50.

[28] D. Patel, S. Lin, J. Rayfield, N. Zhou, C. Shyalika, S. R. Yarrabothula, R. Vaculin, N. Martinez, F. O'Donncha, J. Kalagnanam, Assetopsbench: A real-world evaluation benchmark for ai-driven task automation in industrial asset management, in: Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, ACM, 2026, p. 9649–9660. URL: http://dx.doi.org/10.1145/3770855.3817562. doi:10.1145/3770855.3817562.

[29] M. R. Shadi, H. Mirshekali, H. R. Shaker, Explainable artificial intelligence for energy systems maintenance: A review on concepts, current techniques, challenges, and prospects, Renewable and Sustainable Energy Reviews 216 (2025) 115668. doi:10.1016/j.rser.2025.115668.

[30] X. Zhao, M. Blum, F. Gao, Y. Chen, B. Yang, L. Marquez-Carpintero, M. Pina-Navarro, Y. Fu, S. Morikawa, Y. Iwasawa, Y. Matsuo, C. Park, I. Li, AGENTiGraph: A multi-agent knowledge graph framework for interactive, domain-specific LLM chatbots, in: Proceedings of the 34th ACM International Conference on Information and Knowledge Management, ACM, 2025, pp. 6757–6761.

doi:10.1145/3746252.3761459, demonstration paper.
[31] H. Li, X. Cheng, X. Zhang, Accurate insights, trustworthy interactions: Designing a collaborative ai-human multi-agent system with knowledge graph for diagnosis prediction, in: Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems, 2025. doi:10.1145/3706598.3713526.
[32] G. A. J. Zia, A. Valdestilhas, B. Moreno Torres, S. Kruschwitz, LLM-driven multi-agent inspection planning via semantically enriched knowledge graphs for non-destructive testing, in: Proceedings of SeMatS 2025: 2nd International Workshop on Semantic Materials Science (co-located with ISWC 2025), 2025, pp. 1–13. CEUR Workshop Proceedings, vol. 4104.
[33] M. Kiasari, H. Aly, Agentic artificial intelligence for smart grids: A comprehensive review of autonomous, safe, and explainable control frameworks, Energies 19 (2026) 617. doi:10.3390/en19030617.
[34] T. Yasuno, Quantized vision-language models for damage assessment: A comparative study of llava-1.5-7b quantization levels, arXiv preprint arXiv:2603.26770 (2026).
[35] H. Cai, Y. Rao, L. Huang, Z. Zhong, J. Dong, J. Tan, C. Nai, J. Hou, W. Lu, R. Zhong, Airnav: A large-scale uav vision-and-language navigation dataset with natural and diverse instructions, arXiv preprint arXiv:2601.03707 (2026).