AUGMENTED REALITY FOR KNOWLEDGE DISCOVERY THROUGH FEATURE DETECTION AND SEMANTIC SIMILARITY

Igor KLEIMAN [1*], Marian GHEORGHE[2], Eduard Laurențiu NIȚU[3]

The need for real-time, context-aware knowledge discovery in AR environments led to a new approach that combines feature-based object recognition and semantic similarity search—an innovation beyond current systems. This method uses feature extraction to construct feature graphs, which are matched to knowledge products within knowledge bases. Based on object recognition and contextual cues, relevant knowledge products are retrieved and overlaid on physical objects via Augmented Reality Markup Language. This integration empowers users to access context-specific knowledge in real-time, enhancing decision-making and efficiency. In the context of industrial robotics, it shows substantial improvements in knowledge retrieval and task performance.

Keywords: Augmented Reality (AR), Knowledge Mesh, Data Mesh, Knowledge Product, Knowledge Product Contract, ARML (Augmented Reality Markup Language), XR (Extended Reality), SPARQL (Protocol and RDF Query Language), SIFT (Scale-Invariant Feature Transform), Intelligent Digital Mesh (IDM), Laplacian of Gaussian (LoG), Difference of Gaussians (DoG)

1. Introduction

Augmented Reality-enabled implicit knowledge discovery delivers real-time contextual information through visual recognition of objects, enhancing interaction by automatically overlaying relevant knowledge without explicit searches. Using Scale-Invariant Feature Transform (SIFT**)** for feature extraction; the system detects key points in objects and matches them to stored knowledge products. This integration of AR**,** SIFT**,** and ARML allows users to access 3D models, instructions, or data directly on objects, improving task efficiency and decision-making.

* Corresponding author

[1] Doctorand, Doctoral School of Industrial Engineering and Robotics, POLITEHNICA Bucharest, Romania, OKO Suisse Kleiman, Switzerland, igor.kleiman@stud.fiir.upb.ro

[2] Professor, Dr., Manufacturing Engineering Dept., Doctoral School of Industrial Engineering and Robotics, POLITEHNICA Bucharest, Romania, marian.gheorghe@upb.ro

[3] Professor, Dr., Manufacturing and Industrial Management Dept., Doctoral School of Industrial Engineering, POLITEHNICA Bucharest, Pitești University Centre, Romania, eduard.l.nitu@upb.ro.

[◊] This paper is drawn up to be presented, as appropriate, in a section of ICAMaT 2024

Scale-Invariant Feature Transform (SIFT) [1] [2] is a widely used feature extraction method in computer vision, known for its robustness in object recognition tasks due to its invariance to scale, rotation, and lighting. These properties make it ideal for real-world applications where environmental changes can impact recognition accuracy. However, SIFT has limitations in AR. Its computational complexity can hinder real-time performance, especially in large-scale environments. While effective in static object recognition, SIFT struggles with moving objects or interactive scenarios, areas where AR requires real-time responsiveness. This highlights the need for faster or more efficient methods in feature extraction for AR applications.

In addition to SIFT, other feature extraction techniques have emerged to address specific challenges [3] [4]: Speeded-Up Robust Features (SURF), a faster alternative to SIFT, reduces computation time but may sacrifice some accuracy; Oriented FAST and Rotated BRIEF (ORB), optimized for speed, is popular in mobile AR applications, though less robust in terms of scale invariance; Deep Learning Methods/ Convolutional Neural Networks (CNNs) have increasingly replaced traditional methods, providing higher accuracy in object detection but requiring significant computational resources [4]. While alternatives like SURF and ORB address speed issues, the integration of semantic knowledge retrieval in AR systems remains an open challenge in feature-based recognition.

Implicit Knowledge Discovery in AR-Enabled Systems [5] [6] [7] aims to retrieve knowledge automatically, based on contextual cues, without explicit user input. This approach can enhance AR systems by providing users with relevant information in real time, based on their environment or interaction. Despite its potential, implicit knowledge discovery remains underexplored in AR research. Current systems focus primarily on explicit knowledge retrieval, where users manually search for information. While some AR applications utilize contextual cues, there is a significant gap in integrating semantic similarity search with feature recognition.

The adoption of complex system modeling methodologies has become essential in developing innovative solutions that bridge the gap between digital and physical interactions. Building on the Relational Modeling Framework for Complex Systems [8], which emphasizes the integration of diverse modeling formalisms to handle heterogeneous data and interactions, this research explores an augmented reality (AR) approach to enable real-time, context-aware knowledge discovery.

The concept of the Intelligent Digital Mesh (IDM) [9] [10] has emerged as a key trend in fields like smart manufacturing (SM). IDM represents a digital, interconnected system that links various platforms and devices to create an integrated operational environment. Despite the promise of IDM, its development faces challenges, particularly in data sharing and system integration.

Digitization trends and intense competition in the industry require universities to innovate in production methods and techniques while providing graduates with the specific skills needed for their application. The emergence of Learning Factories and Industry 4.0 has fostered the development of techniques tailored to industrial processes and systems [11] [12]. These techniques include virtual, augmented, and mixed reality, as well as digital assistance, project-based learning, problem-based learning, and intuitive learning.

Augmented reality blends the real and virtual worlds through real-time interactions, precise registration of actual and virtual 3D objects, and their superimposition in the real environment.

This paper is introducing a system that merges SIFT-based feature recognition, semantic similarity search, and ARML, enabling real-time, context-aware knowledge discovery in AR environments. It propagates a novel solution that leverages AR Markup Language (ARML) and implicit knowledge discovery to create a context-aware knowledge retrieval system for Intelligent Digital Mesh (IDM) environments, particularly addressing gaps in smart manufacturing. The core innovation integrates Scale-Invariant Feature Transform (SIFT) for feature extraction, semantic similarity search to match feature graphs with knowledge products, and ARML to overlay those products in real-time onto physical objects.

2. Modelling on feature graphs

The Scale-Invariant Feature Transform (SIFT) detects key points through a multi-step process that begins with constructing a scale-space representation, where the image is progressively blurred at different scales to identify stable features. Using the Difference of Gaussian (DoG) technique, local extrema are detected as potential key points across scales. Each key point is then refined to ensure contrast and localization accuracy. Finally, SIFT assigns orientations to each key point based on local gradients, generating descriptors that capture the local gradient distribution around each key point. These descriptors are invariant to scale and rotation, making them effective for consistent recognition across varied conditions.

To generate the Keypoint Descriptor (Feature Vector) in SIFT, we focus on creating a representation that captures the local gradient information around each detected keypoint, as follows.

1. Define the Keypoint Region: around each keypoint, a square region (typically 16x16 pixels) is selected for analysis; this region provides the context for the keypoint and is subdivided into a 4x4 grid of smaller cells (each cell being 4x4 pixels).
2. Compute Gradient Orientations and Magnitudes: for each pixel in the 16x16 region, gradients are calculated by examining the changes in intensity along the x and y directions; this gives both a magnitude (the strength of the gradient) and an orientation (the direction of the gradient) at each pixel.

3. Orientation Histogram for Each Cell: for each of the 4x4 cells within the keypoint region, an 8-bin orientation histogram is created; each bin represents a 45-degree range (covering a total of 360 degrees); pixels within each cell contribute to the histogram based on their orientation, with the gradient magnitude used as the weight for the histogram bins.
4. Normalize and Concatenate Histograms: the histograms from each of the 16 cells (4x4 grid) are concatenated to form a 128-dimensional feature vector (4x4 cells × 8 bins), as shown in Fig. 1; this vector is then normalized to reduce the effects of lighting and contrast variations, ensuring the descriptor's robustness.

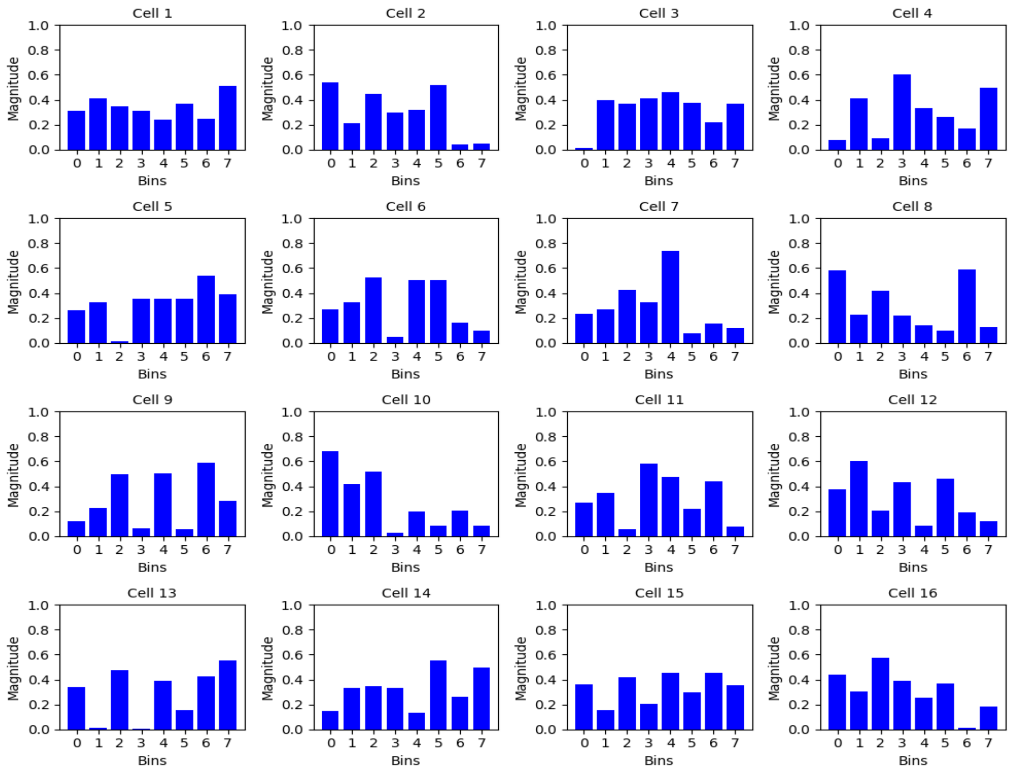

Fig. 1. Normalized Orientation Histograms for SIFT Keypoint Descriptor
(4x4 cells, 8 bins each)

While SIFT provides robust key point detection and descriptive features, this article builds on its foundation by introducing an innovative approach: the feature graph. Rather than using key points solely for direct matching, the feature graph leverages SIFT-extracted key points and organizes them into a structured graph format. In this framework, each key point serves as a node, while edges represent spatial and orientation relationships between these nodes, creating a structural "fingerprint" of the object.

After key points are extracted from the image, the system organizes them into a feature graph, where the key points are treated as nodes and the relationships between them (such as their spatial arrangement) are represented as edges.

This graph encapsulates the structural representation of the object. The feature graph serves as a unique fingerprint of the object, making it possible to match it with pre-existing graphs stored in a knowledge base.

A feature graph is:

$$G = (V, E) \tag{1}$$

where: V is the set of nodes (key points) and E - the set of edges, which describe relationships between nodes (key points).

A node, v_i, is defined by the followings:

$$v_i \in V,\ v_i = (x_i, y_i,\ \theta_i,\ s_i, d_i) \tag{2}$$

where: x_i and y_i are the coordinates of the key point, $\theta_i -$ dominant orientation of the local image gradient information around the key point, $s_i - s$cale at which the key point was detected, d_i - the 128 - dimensional SIFT descriptor vector.

An edge e_{ij} between nodes v_i and v_j is defined by the followings:

$$e_{ij} \in E, (v_i\ e_{ij}\ v_j),\ e_{ij} = \left(d_{ij}, \Delta\theta_{ij}\right) \tag{3}$$

$$d_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \tag{4}$$

$$\Delta\theta_{ij} = \left|\theta_i - \theta_j\right| \tag{5}$$

were: d_{ij} represents Euclidean distance between the key points of coordinates (x_i, y_i) and (x_j, y_j); $\Delta\theta_{ij}$ - the difference between orientations θ_i and θ_j of the local image gradient information around each detected key point pair (x_i, y_i) and (x_j, y_j).

3. Case Study

In our R&D project, we utilized the Google Cloud Platform Architecture for an AR and Feature Graph-Enabled Knowledge Mesh.

The sequence diagram illustrates the collaborative interaction between system components in facilitating real-time, context-aware knowledge retrieval through AR, as shown in Fig. 2.

Beginning with the SIFT-based component identification, the AR device captures key points and sends the generated feature graph to the Neo4j Knowledge Base.

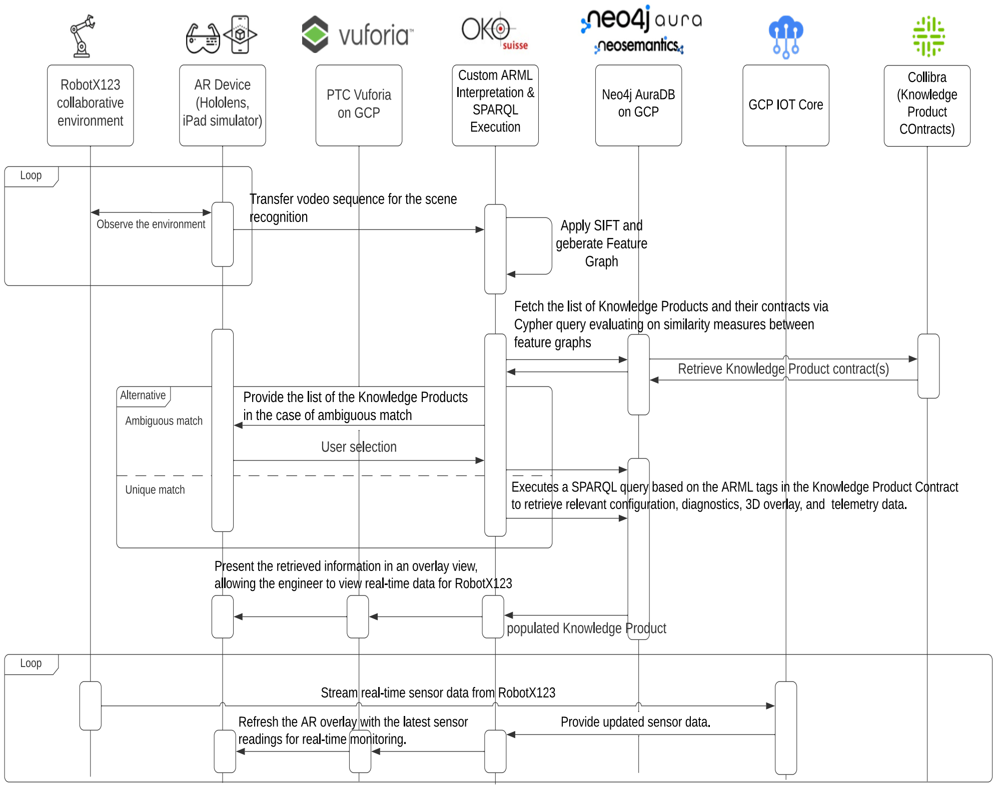

Fig. 2. Sequence Diagram: Feature Graph - Enabled Knowledge Retrieval Workflow

Through sub-graph matching, Neo4j utilizes Cypher queries to locate relevant knowledge products, even allowing partial matches to improve retrieval accuracy. Once identified, these knowledge products are presented through the AR interface, with the VR interface assisting in ambiguity handling if multiple documents match. Finally, the selected knowledge product is delivered as an overlay on the AR device, enabling seamless collaboration between AR, feature detection, and database components to support efficient knowledge discovery.

The system employs SIFT for object recognition, automatically identifying machine components by extracting their feature graphs. This information is used as a composite identifier to search a knowledge base (stored in a graph database like Neo4j), where relevant engineering documentation (e.g., configuration guides, calibration protocols) is stored and indexed by feature graphs, as follows.

(1) Component Identification with SIFT

- The engineer uses an AR headset or AR-enabled tablet to scan the machine component.

- The system runs SIFT (Scale-Invariant Feature Transform) to detect key points on the component, creating a feature graph representing the component's unique structure. The feature graph is embedded into ARML as presented in Fig. 3.

```
KnowledgeProductContract:
  knowledgeProductId: "KP12345"
  name: "Machine Maintenance Guide"
  version: "1.0"
  ARML:
    Location:
      Latitude: 37.7749
      Longitude: -122.4194
    3DModel:
      ModelURL: "http://example.com/models/machine123.glb"
      Placement:
        Orientation: "0,0,0"
        Scale: 1.0
    Metadata:
      KnowledgeProduct: "MaintenanceGuide"
      Trigger: "Proximity"
    SIFTFeatureGraph:
      KeyPoints:
        - {x: 123.4, y: 567.8, scale: 1.6, orientation: 0.1, descriptor: "0.13,0.45,0.32,..."}
        - {x: 223.5, y: 667.9, scale: 1.2, orientation: 0.15, descriptor: "0.21,0.34,0.31,..."}
        - {x: 323.6, y: 767.0, scale: 1.0, orientation: 0.2, descriptor: "0.22,0.47,0.35,..."}
      Relationships:
        - {sourceKeyPoint: 123.4, targetKeyPoint: 223.5, distance: 142.07}
        - {sourceKeyPoint: 223.5, targetKeyPoint: 323.6, distance: 120.34}
      Detection:
        - FeatureDetected: "SIFT-based detection of component"
        - ProductId: "Product123"  # Dynamically identified product from feature graph matching
        - ActionOnDetection:
            SPARQLQuery:
              query: |
                SELECT ?maintenanceGuide
                WHERE {
                  ?product rdf:type ex:Product .
                  ?product ex:hasInstruction ?maintenanceGuide .
                  FILTER (?product = <http://example.com/ontology#{{ProductId}}>)
                }
              parameters:
                - ProductId: "{{FeatureGraphMatchedProductId}}"
      Visualization:
        Description: "When the SIFT feature graph detects the component, this dynamically triggers
```

Fig. 3. SIFT feature graph embedded into ARML

(2) Feature Graph Search in the Knowledge Base

- The extracted feature graph is used to perform a sub-graph matching search in the Neo4j knowledge base. The knowledge base contains a library of knowledge products, each associated with a stored feature graph, as shown in Fig. 4.
- Sub-graph matching ensures efficient retrieval of relevant knowledge products, even when the match is not exact. This improves performance by allowing partial matches to retrieve the closest results. Fig. 5 illustrates the

Cypher query identifies and retrieves a Knowledge Product in a Neo4j graph database based on key point relationships, using similarity measures between descriptors.

```
CREATE (kp1:KeyPoint {x:100, y:150, orientation:45, scale:1.5,
descriptor:"0.22,0.45,..."})
CREATE (kp2:KeyPoint {x:200, y:250, orientation:90, scale:1.8,
descriptor:"0.13,0.55,..."})
CREATE (kp1)-[:RELATED {distance:142.07, angle_difference:45}]->(kp2)

CREATE (kp2)-[:IDENTIFIES]->(knowledgeProduct:KnowledgeProduct {id:"KP123",
name:"Maintenance Guide"})
```

Fig. 4. Index the Feature Graph as an Identifier for the Knowledge Product

```
MATCH (newKP1:KeyPoint {descriptor:"0.22,0.45,..."})-[:RELATED
{distance:142.07}]->(newKP2:KeyPoint {descriptor:"0.13,0.55,..."})

MATCH (newKP1)-[:IDENTIFIES]->(knowledgeProduct:KnowledgeProduct)

RETURN knowledgeProduct
```

Fig. 5. Neo4j Cypher query for knowledge retreaval based on similarity measures between feature graphs

(3) Knowledge Product Retrieval and Ambiguity Handling

- If a clear match is found, the system retrieves the corresponding knowledge product—such as a maintenance guide, operational history, or real-time sensor data.
- If the match is ambiguous (multiple potential documents are returned), the engineer is presented with multiple AR overlays, each corresponding to a different document. Using the Virtual Reality (VR) interface, the engineer can interact with the AR representations of the knowledge products, preview them, and select the correct one for the task at hand.

(4) AR Overlay for Context-Aware Knowledge Delivery

Once the correct knowledge product is selected, the system overlays the relevant information directly onto the machine component. This may include:

- configuration guidelines with step-by-step instructions;
- calibration data and real-time metrics such as pressure, alignment, or torque values;
- operational parameters and safety thresholds required for the component.

A demonstration of AR-enabled Knowledge Mesh solution in a live setting showcasing real-time interaction with a model of an industrial pump unit using augmented reality is as presented in Fig. 6.

Fig. 6. Live Demonstration of AR-Enabled Knowledge Mesh Solution for Real-Time industrial device interaction

4. Conclusions

The presented system demonstrates the potential for AR-enabled implicit knowledge discovery by seamlessly integrating feature-based recognition with semantic similarity search. This approach facilitates real-time access to contextually relevant knowledge, essential in complex industrial settings. By leveraging SIFT-based feature graphs and AR Markup Language (ARML), the solution enables a unique identification process for knowledge products that boosts operational efficiency and decision-making.

The application of feature graphs as unique identifiers within the Knowledge Mesh framework proves versatile, with scalable potential across diverse fields such as robotics, engineering, and quality assurance. By enabling AR-based visualization, this approach supports engineers in accessing critical information directly in the physical environment, bridging the gap between data and its real-world application. While effective, the system's reliance on SIFT presents challenges, particularly in real-time performance due to SIFT's computational intensity. The need for optimized feature matching methods, like sub-graph matching algorithms and hardware acceleration, becomes apparent to harness AR-driven knowledge discovery in dynamic environments.

Future research should explore deep learning-based feature extraction and enhanced matching techniques to improve efficiency in larger, complex environments, expanding multi-object recognition and collaborative AR capabilities to push the boundaries of AR-enabled knowledge management.

REFERENCES

[1] *R. Doerner, W. Broll, P. Grimm, B. Jung,* Virtual and augmented reality (VR/AR): Foundations and methods of extended realities (XR), Cham, Switzerland, Springer, 2022, p. 429.

[2] *G. Lowe*, SIFT - The Scale Invariant Feature Transform., International Journal, Nr. 2, 2004.

[3] *X. Yang, Y. C. Lu, A. Murtiyoso, M. Koehl, P. Grussenmeyer,* HBIM modeling from the surface mesh and its extended capability of knowledge representation, ISPRS International Journal of Geo-Information, Bd. 8, Nr. 8(7), 2019.

[4] *L. V. Lozano-Vázquez, J. Miura, A.J. Rosales-Silva, A. Luviano-Juárez, D. Mújica-Vargas*, Analysis of Different Image Enhancement and Feature Extraction Methods, Mathematics, Bd. 10, Nr. 14, 2022.

[5] *G. Dragoi, A. Draghici, S. M. Rosu, A. Radovici, C. E. Cotet,* Knowledge base development in virtual enterprise network as support for workplace risk assessment, International Journal of Human Capital and Information Technology Professionals, Bd. 2, Nr. 3, 2011.

[6] *K. Iskandar, H. Prabowo, R. Kosala, A. Trisetyarso,* A framework for knowledge management system with mapreduce approach to overcome information overload, ICIC Express Letters, Bd. 13, Nr. 10, 2019.

[7] *T. Bai, L. Gong, C.A. Kulikowski, L. Huang,* Implicit knowledge discovery in biomedical ontologies: Computing interesting relatednesses, in IEEE International Conference on Bioinformatics and Biomedicine, BIBM, 2015.

[8] *D.C. Popescu, I. Dumitrache*, Relational modeling framework for complex systems, U.P.B. Sci. Bull., Series *C,* V. **83**, Nr. 1, 2021.

[9] *M.P. Uysal, A.E. Mergen,* Smart manufacturing in intelligent digital mesh: Integration of enterprise architecture and software product line engineering, Journal of Industrial Information Integration, Bd. 22, 2021.

[10] *B. Nedić,* Gartner's top strategic technology trends, Proceedings on Engineering Sciences, Bd. 1, Nr. 2, 2019.

[11] *E.L. Nitu, A.C. Gavriluta,* Lean Learning Factory at the University of Pitesti. In Modern Technologies in Industrial Engineering VII, IOP Publishing: Bristol, UK, V. **591**, Nr. 1, 2019.

[12] *G.C. Neacşu, E.L. Niţu, A.C. Gavriluţă, G.G. Vlad, E.M. Dobre, M. Gheorghe, M.M. Stan,* Process Analysis and Modelling of Operator Performance in Classical and Digitalized Assembly Workstations, Processes 2024, 12, 533.