*Athens Journal of Technology & Engineering 2025, 12: 1-15*
https://doi.org/10.30958/ajte.X-Y-Z

# Enhancing Multimodal Systems using Azure AI Image Embeddings and GPT-4.x Vision

*By Mihail Mateev*[*]

*Image embeddings have become foundational in AI, enabling machines to transform visual data into structured numerical vectors for applications from predictive analytics to interactive user experiences. This paper presents a comprehensive study of Azure AI's image embedding technologies and their integration with GPT-4.x Vision to enhance multimodal retrieval-augmented generation (RAG) systems. We analyze the comparative strengths of Azure Machine Learning custom pipelines, the Azure AI Model Inference API, and Computer Vision v4.0 multimodal embeddings. The integration of CLIP embeddings and hybrid decomposition strategies is explored, with empirical results from a predictive maintenance case study. Theoretical frameworks, solution architectures, and experimental results are illustrated with six original schemas and diagrams, providing practical guidance for deploying scalable, accurate, and cost-efficient multimodal AI systems.*

**Keywords:** *ChatGPT, AI, Generative AI, AI Foundry, Vector Search, Open AI, Azure AI Vision, Microsoft Azure, Image Embeddings, Predictive Analysis, IoT, Power Platform*

## Introduction

The development of large language models (LLMs) and computer vision has changed how organizations utilize visual data. Image embeddings—dense vector representations of images—enable efficient search, classification, and cross-modal analytics. The ability to align image and text data in a unified vector space underpins advanced multimodal systems, enabling new applications in predictive maintenance, security, medical diagnostics, and e-commerce (Mateev 2025, Mateev 2024, Mateev 2024).

Azure AI, with its suite of embedding and vision services, offers a robust platform for deploying such systems at scale. This paper extends previous analyses by providing a detailed comparison of Azure's embedding services, integrating theoretical and practical perspectives, and presenting empirical results from real-world deployments.

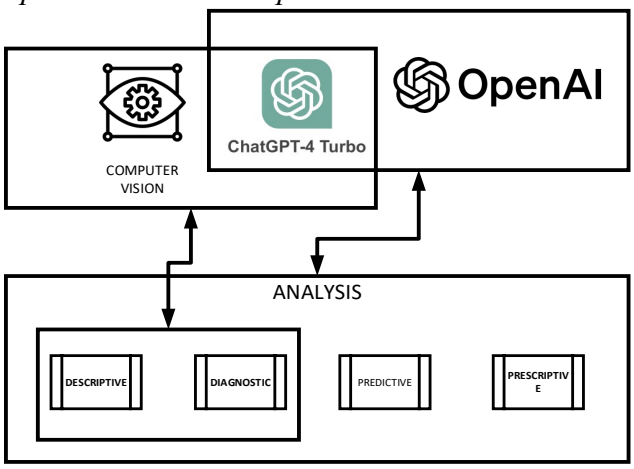
This study explores image analysis with Generative AI and GPT models via OpenAI API. OpenAI is a relatively new organization and laboratory for artificial intelligence research. It was established in 2015 by several tech leaders, such as Elon Musk and Sam Altman. Artificial Intelligence is the research field that OpenAI works in across different areas such as natural language processing (NLP), computer vision, reinforcement learning, and robotics.

[*]Chief Assistant Professor, UACEG, Faculty of Structural Engineering, CAE Department, Bulgaria.

The current research focuses on developing a reference architecture for leading image and video analysis platforms that utilize image embeddings. The focus is on various LLMs, which provide image analysis and image embeddings. The research uses preliminary OpenAI and Cohere models, GPT models, enhancing the analysis with computer vision services - especially GPT-4.x with Vision (GPT4-TV) to implement with a short time to market solutions, able to be implemented fast, to provide accurate results and to be available also for low code /no code technologies (Mateev 2025).

The paper extends the research from (Mateev, 2025), where options to use together GenAI and Computer Vision. The overall schema explaining relations between different types of analysis using Computer Vision and Open AI is demonstrated in Figure 1.

**Figure 1.** *Implementation of Different Types of analysis using Computer Vision, Open AI – General Dependencies*




The next chapter covers details on the research background and related work.

## Methodology

*Concept, Technologies, and Methodology Description*

The Role of Image Embeddings

Image embeddings are high-dimensional vectors generated by deep learning models, capturing the semantic and contextual features of images. These vectors enable:

- **Similarity search**: Finding visually or semantically similar images.
- **Multimodal retrieval**: Linking images and text for cross-modal search.
- **Clustering/classification**: Grouping images by content or features.
- **RAG systems**: Enhancing LLMs with contextual image data for more accurate generation (Mateev 2025, Mateev 2024).

Multimodal Systems and RAG

Multimodal systems integrate data from multiple sources (e.g., images, text, audio) to provide richer, context-aware outputs. Retrieval-augmented generation (RAG) leverages embeddings to retrieve relevant content, which is then used by LLMs like GPT-4.x Vision for reasoning and generation (Mateev 2025, Mateev 2024).

Azure AI Image Embedding Solutions

In this section, options for Image Embeddings implementations are considered, available on the cloud platform used for this research: Microsoft Azure.

Azure Machine Learning Custom Embedding Pipelines

It is possible to create a custom AI service (based on AML or another AI platform) that implements custom AI algorithms. Some open-source projects and libraries can be utilized for developing custom embedding modules.

- **Customizability**: Allows domain-specific tuning.
- **Batch processing**: Efficient for large datasets.
- **Integration**: Seamless with Azure ML workflows (Mateev, 2025).

Azure AI Model Inference API

This option is the preferred one for research, offering simplicity and access to the latest LLMs as a service.

- **Pre-trained models**: Rapid deployment for general-purpose use.
- **Unified API**: Simplifies integration and model switching.
- **Scalability**: Supports serverless and managed compute endpoints (Mateev 2025).

Computer Vision v4.0 Multimodal Embeddings

- **Unified vector space**: Integrates images and text for cross-modal search.
- **Multilingual support:** Over 100 languages for text queries.
- **Optimized for RAG**: Enhances retrieval and generation tasks (Mateev 2025, Mateev 2024).

Integration with CLIP

CLIP (Contrastive Language-Image Pre-training) aligns images and text in a shared space, enabling zero-shot learning and flexible retrieval. Integration with Azure enables enhanced multimodal RAG systems (Mateev 2025, Mateev 2024).

## The Theoretical and Technical Framework

The experimental research environment uses ChatGPT-4. x with Vision (ChatGPT-4-TV) for image analysis. This model has the following specifics:

- Open AI-based
- LLM-based.
- Good at solving not defined requirements in both conversational and completion modes
- Option to use Retrieval Augmented Generation (RAG)

In the case related to analyzing the video stream (input stream from drones and robots' cameras during construction observation), Azure AI Vision empowers ChatGPT-4-TV.

Azure AI Vision is a distinct solution for computer vision analysis, utilizing cognitive analysis to comprehend videos and images. It has the following characteristics:

- It's not LLM-based.
- Good at clearly defined tasks, object recognition, etc.
- Working with video streaming.

AI Vision extracts frames from the stream and sends them to the OpenAI service using a ChatGPT-4. x deployment.

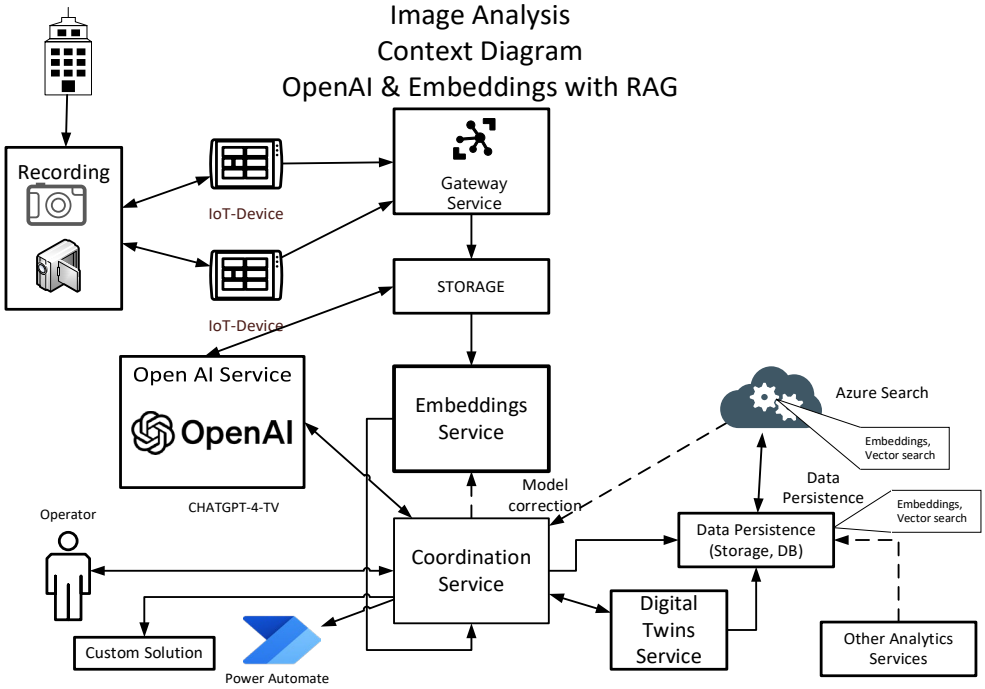
This study is focused on exploring the impact of various innovative approaches to improving efficiency and reducing costs in the realm of image analysis, particularly through the application of OpenAI's advanced multimodal language models, such as GPT-4-Turbo with Vision (ChatGPT-4-TV) and GPT-4. These models are deployed to perform a range of complex analytical tasks that go beyond traditional methods.

The experimental setup includes a sophisticated module for case decomposition, designed with the capabilities of Azure Digital Twins (ADT). This module is integral to the process, as it organizes and supervises the contextual framework of the solutions under analysis. It effectively breaks down intricate cases into smaller, manageable components, ensuring seamless integration of analyses. The decomposition process transforms domain-specific problems into general subcases, which are then analyzed using the robust tools provided by these large language models. By leveraging this systematic approach, the study aims to unlock new potential in image analysis, enhancing its applicability and precision across diverse scenarios.

One high-level schema of the solution is presented in the schema below.

Figure 2 illustrates the contextual schema of a solution utilizing OpenAI and AI Vision for digital content analysis.

**Figure 2.** *Predictive Analysis using Image Analysis with OpenAI and Embeddings*




The experimental framework employed for this study is meticulously designed using Microsoft Azure, Azure OpenAI Services, and Power Automate. This setup incorporates a diverse dataset, which includes 2,000 high-resolution images alongside 10 concise video clips, each with a duration of up to 20 seconds. The visual data primarily focuses on corroded steel construction elements sourced from actual structures within the United States. Figure 3 illustrates this dataset, providing rich visual references that form the foundation of the analysis process.

**Figure 3.** *Images from the Experimental Test Set: Steel Construction with Corrosion*

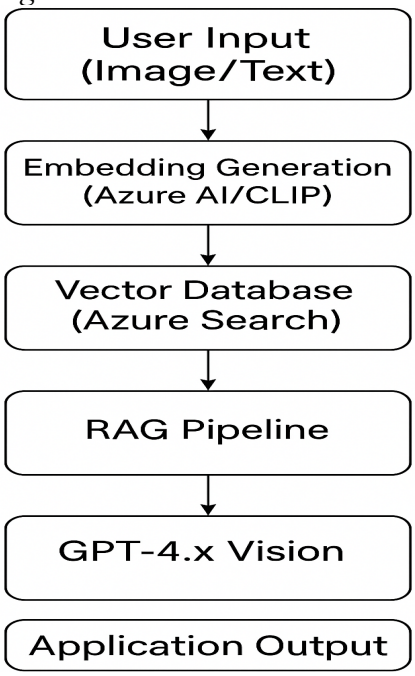
*Hybrid Decomposition Architecture*

The image analysis usually requires several steps, including:

- Collect the content (images)
- Generate embeddings
- Persist the embeddings in a vector database
- Search the distance with other embeddings in the database (RAG pipeline)
- Request a solution from AI (could be in several steps using LLMs and computer vision in different steps)
- Output results (for the user and persist them in a database)

Complex tasks are decomposed into smaller subtasks using the principles of Separation of Concerns (SoC) and Digital Twins (DT). This enables parallel processing and efficient resource utilization (Mateev, 2025; Mateev, 2024).

**Figure 4.** *High-Level Multimodal Solution Architecture (main flow)*




Processing chain details: The main flow begins with image/video ingestion, followed by on-the-fly frame extraction (when applicable), preprocessing (resize/ normalize), and embedding generation (Azure AI Vision or CLIP). Embeddings are persisted in a VectorDB (e.g., Azure AI Search / FAISS-compatible store) with metadata (timestamp, source, camera, asset id). Queries—text or image—undergo the same embedding function to ensure space alignment. Top-k candidates are retrieved via cosine similarity and passed as structured context into the LLM stage (GPT-4.x Vision) for reasoning, explanation, and decision support. The response and relevant matches are logged for traceability and future retraining.

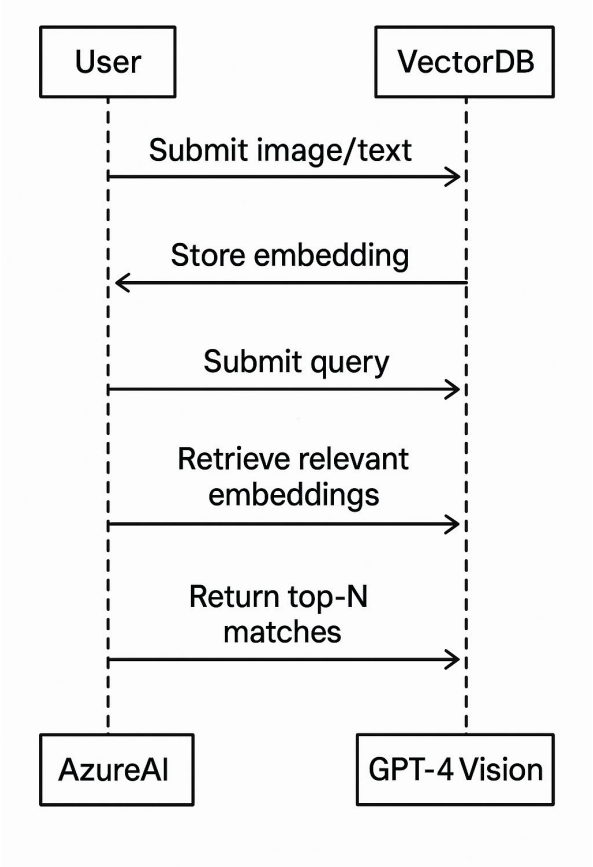
*Embedding and Vector Search Workflow*

RAG Pipeline Processing Details

The sequence diagram in Figure 5 below illustrates the interaction between four entities: User, AzureAI, VectorDB, and GPT-4 Vision. The user submits image or text data to AzureAI, which processes the input and stores its embedding in VectorDB. Later, the user sends a query to GPT-4 Vision, which retrieves the most relevant embeddings from VectorDB and returns the top-N matching results.

- Indexing: (i) Normalize images; (ii) produce embeddings; (iii) store vectors + metadata; (iv) build/update ANN indexes (HNSW/IVF).
- Retrieval: Convert user image/text to an embedding and perform vector search (cosine/L2) with filters (time/window, asset type, confidence).
- Augmentation: Package top-k results (thumbnails, captions, scores) as context, optionally merged with domain documents (manuals, SOPs) via hybrid (BM25+vector) search.
- Generation: GPT-4.x Vision composes an answer/explanation, citing retrieved items and highlighting uncertainty thresholds.
- Feedback & Logging: Capture user confirmations/corrections to update relevance signals and refresh the index.

**Figure 5.** *Embedding and Vector Search Pipeline*

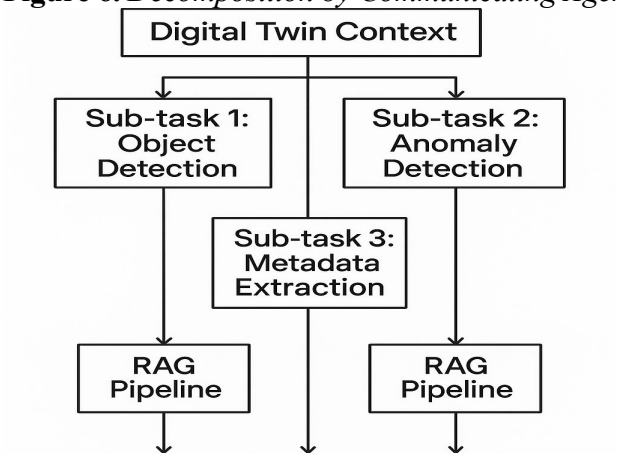


Often, complex flows need to be split into a system of simpler flows, and for such scenarios, the approach used is based on decomposition using the DT (Digital Twins) concept. The design, based on DT, simplifies the logic of the solution and allows for applying more formal analysis (decoupled from the context) for part of the flows.

*Decomposition for Complex Image Analysis*

The diagram illustrates a decomposition of tasks within a Digital Twin Context using communicating agents. The central context is broken down into three sub-tasks: Object Detection, Anomaly Detection, and Metadata Extraction. Each sub-task acts as an independent agent that processes specific information. These agents communicate their results to corresponding RAG Pipelines for further reasoning and retrieval-augmented generation. This modular design enables scalable, context-aware decision-making throughout the digital twin system.

**Figure 6.** *Decomposition by Communicating Agents*




Within the realm of image analysis and image embeddings, this workflow illustrates a modular and task-specific architecture designed for processing and comprehending intricate visual data in the context of a "Digital Twin system":

1. **Digital Twin Context**: Represents the virtual replica of a physical system (e.g., a smart building or industrial environment) where real-time data from sensors or cameras is continuously fed.
2. **Sub-Task Agents**:

    - **Object Detection**: Identifies and classifies entities (e.g., equipment, people, vehicles) within images. It generates localized features and object labels.
    - **Anomaly Detection**: Detects deviations from normal patterns using visual embeddings—useful in predictive maintenance or safety monitoring.

- **Metadata Extraction**: Captures contextual information from the image (e.g., timestamps, location, visual descriptors) and converts it into structured data or embeddings.

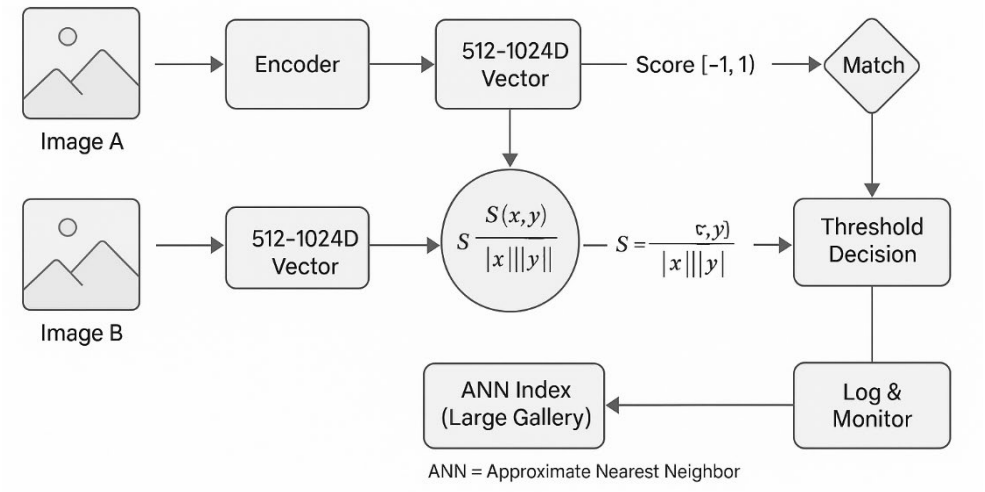
3. **RAG Pipelines** (Retrieval-Augmented Generation): Each sub-task sends its embeddings to a dedicated RAG pipeline, where relevant knowledge (e.g., documents, technical manuals, or prior cases) is retrieved from a vector database. This allows the system to generate rich, informed responses or alerts based on both the visual input and external knowledge.

The application of such a solution could be in various domains: for example, in a factory setting, this architecture could analyze surveillance footage in real-time, detect machinery, flag overheating components, extract operational context, and query technical documentation to provide instant insights or automated reports. The use of embeddings and retrieval makes it scalable and explainable.

*Embedding Generation and Similarity Computation*

The schema shows how two images are converted into embeddings, which are then compared using cosine similarity. This comparison calculates a similarity score indicating how visually or semantically alike the two images are. It's a key process in image matching, retrieval, and clustering tasks.

**Figure 7.** *Embedding Generation and Cosine Similarity*



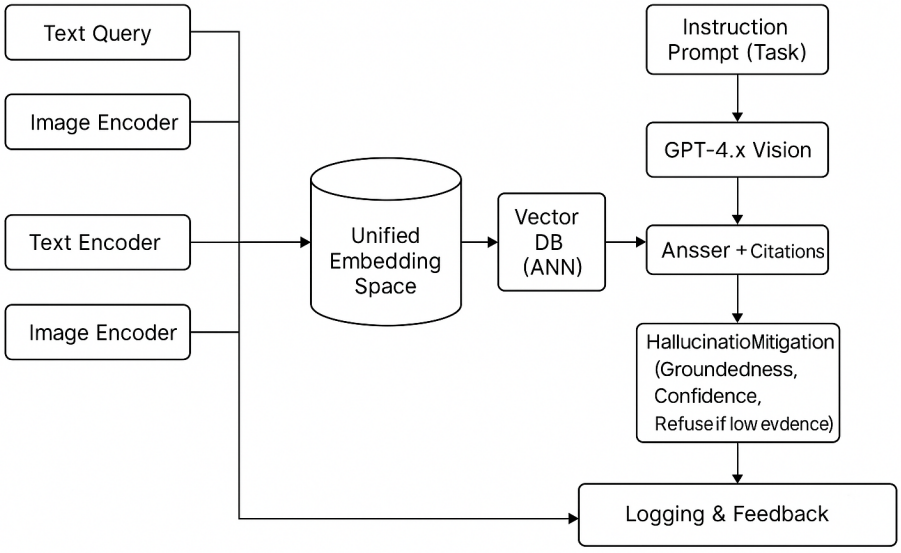

Processing chain details: Two images are independently encoded to 512–1024-D vectors. Cosine similarity $S(x,y)=\langle x,y\rangle/(\|x\|\|y\|)$ yields a score in $[-1,1]$, thresholded for match decisions. Batch normalization of vectors and approximate-nearest-neighbor (ANN) indexes improve both stability and latency for large galleries.

*RAG with Multimodal Embeddings*

Nowadays, most of the LLMs are multi-content. The same trend is promoted for models offering embeddings, such as those that embed different types of content or even models that generate content and provide embeddings.

The diagram illustrates a RAG (Retrieval-Augmented Generation) workflow using multimodal embeddings. A user submits a query—image or text—which is processed through Azure AI Search to perform a vector similarity search. Relevant embeddings are retrieved and sent to GPT-4.x Vision. The model uses them as context to generate a rich, informed output. This approach enhances answer accuracy and relevance by combining LLM generation with retrieved visual or textual knowledge.

**Figure 8.** *Retrieval-Augmented Generation with Multimodal Embeddings*




Processing chain details: Multimodal RAG first retrieves visually and textually relevant items using a unified vector space, then conditions GPT-4.x Vision with (a) the query, (b) retrieved image/text snippets, and (c) task instructions (e.g., defect classification). This reduces hallucinations and improves faithfulness compared with generation-only baselines.

## Research

The experimental environment for this study is based on Microsoft Azure and Azure AI Foundry (a part of the platform that offers AI services)

The research setup employs Azure AI, OpenAI GPT-4: x Vision, and Power Automate for workflow orchestration. The test dataset consists of 10,000 images and 10 short videos of corroded steel structures. Digital Twins are utilized for context management and decomposition, enabling domain-agnostic subcases for Large Language Model (LLM) analysis (Mateev 2025, Mateev 2024, Mateev 2024).

The experimental environment uses ChatGPT 4o with Vision (ChatGPT-4-TV) for image analysis. The solution includes a Digital Twin (DT) module based on Azure Digital Twins (ADT). ADT is used to extract the context from the observed solution, decompose the case, and unify the analysis case, converting it from domain-specific to domain-agnostic subcases suitable for LLM analysis.

One high-level scheme of the solution is presented in Figure 2 (Mateev, 2023).]

## Findings/Results

In this paper, there are added metrics related to the experimental project PoC using the following KPI:

1. Embedding Performance
    a. Azure AI Vision and GPT-4o
    b. Hugging Face
    c. Replicate
    d. ChatGPT-4+CLIP
2. Vector Search Performance
3. Predictive Analysis Accuracy

### *Embedding Performance*

The summarization of results is shown in the Table 1 below.

**Table 1.** *Embedding Performance*

| Service | Embedding Quality | Dimensionality | Inference Speed | Cost Efficiency | Flexibility |
|---|---|---|---|---|---|
| **Azure AI Vision (GPT-4o)** | 9/10 | 512 | Moderate | Moderate | High |
| **Hugging Face** | 8/10 | 768 | Fastest | Best | Highest |
| **Replicate** | 7/10 | 1024 | Slowest | Low | Moderate |
| **ChatGPT-4o (CLIP)** | 9/10 | 512 | Moderate | Moderate | High |

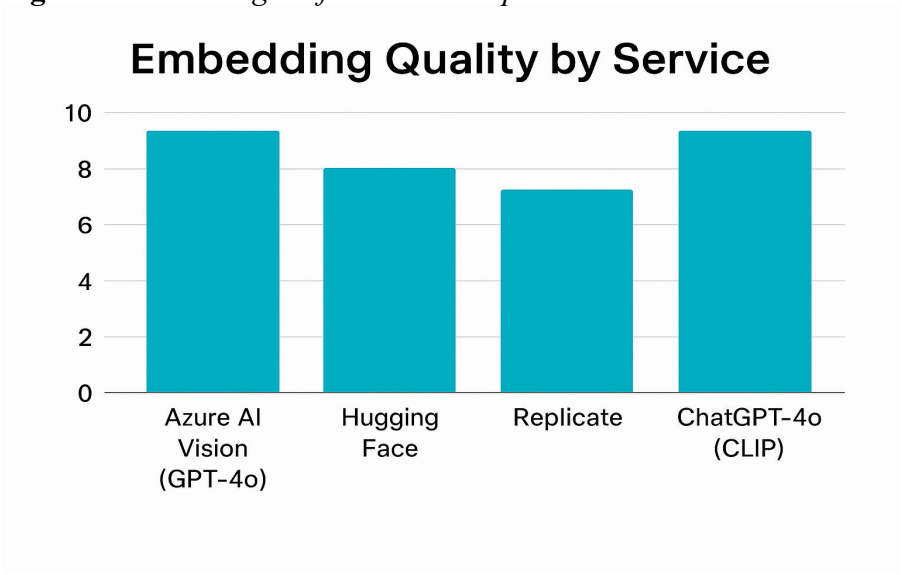
Embedding quality (rated 0–10) reflects how well a model converts inputs (images or text) into vector representations that capture semantic meaning and content similarity. This score is typically based on the following metrics:

Discussion: Hugging Face and Replicate refer to curated model hubs used in our PoC. Hugging Face experiments relied primarily on open-source CLIP variants with ViT backbones that offered strong zero-shot retrieval and fast inference on commodity GPUs. Replicate runs leveraged larger embedding models with 1024-D outputs, which increased storage and latency but provided richer feature representations for difficult edge cases. Vector dimensionality directly impacts index memory and query time; thus, 512-D models tended to be more cost-efficient at scale, while 768–1024-D models were beneficial when maximum recall was required. These trade-offs informed the design recommendations.

- Metrics and protocol: We report Recall@k ($k \in \{1,5,10\}$), MRR, and Precision for retrieval; p50/p95 latency and QPS for serving; and index footprint (GB) at various gallery sizes. Predictive maintenance accuracy is computed per-image and per-case, with confidence-threshold analysis; we additionally track false-positive/negative rates for anomaly detection sub-tasks. These measurement definitions are now included to support reproducibility.
- Zero-Shot Performance: Ability to generalize to unseen queries or tasks, often tested on benchmark datasets (like ImageNet or LAION).
- Cross-Modal Alignment: For multimodal models (e.g., GPT-4o, CLIP), how accurately do image and text embeddings align in a shared space?
- Clustering Consistency: How tightly similar items cluster and how distinctly different items are separated in embedding space (visualized via t-SNE or UMAP).
- Cosine Similarity Tests: Controlled comparisons between image pairs to measure the precision of similarity scores.

The 0–10 scale is often derived from normalized benchmark results or expert evaluations across these categories. A score of 9–10 implies state-of-the-art performance in aligning embeddings with real-world meaning.
The embedding quality comparison is described in Figure 9.

**Figure 9.** *Embedding Performance Comparison*




Vector Search Performance

The Vector Search performance, split by components, is described below:

- Relevance: Azure AI Vision and ChatGPT-4o (CLIP) yield the most relevant results.
- Latency: Hugging Face is fastest; Replicate is slowest.
- Storage: Lower dimensionality (512) is more efficient for storage and computation.
- Cost: Hugging Face is the most cost-efficient; Replicate is the least (Mateev 2025, Mateev 2024).

**Predictive Analysis Accuracy**

The relevance of the services compared by accuracy is described below:

- Azure AI Vision (GPT-4o) and CLIP (ChatGPT-4o) achieve the highest accuracy for predictive maintenance (up to 99.4% in defect detection).
- Hybrid approaches (using SoC and DTs) further optimize performance and cost, especially for complex or large-scale deployments (Mateev 2025, Mateev 2024).

## Conclusions

Azure AI's embedding solutions, combined with GPT-4.x Vision and CLIP enable scalable, accurate, and cost-efficient multimodal systems. The choice of embedding service should be tailored to the application's quality, speed, and budget requirements. Future research will focus on automated decomposition, expanding hybrid frameworks, and maximizing API aggregation platforms for embedding generation (Mateev 2025, Mateev 2024).

*Key Insights*

The central key insight from this research is based on a comparison between the analyzed methods for image embedding generation:

- **Best Quality**: Azure AI Vision and ChatGPT-4o (CLIP) for high-stakes applications.
- **Fastest**: Hugging Face for real-time, large-scale deployments.
- **Most Cost-Efficient**: Hugging Face for startups and budget-conscious projects.
- **Richest Embeddings**: Replicate for research and feature-rich scenarios.

*Design Recommendations*

A critical takeaway from this study is about the design recommendations when implementing image embedding.

- Use custom pipelines for domain-specific tasks.
- Leverage pre-trained APIs for rapid deployment.
- Employ multimodal embeddings for unified search and retrieval.
- Integrate hybrid decomposition for complex workflows.
- Aggregate APIs (e.g., Eden AI) for cost and flexibility optimization(Mateev, 2025; Mateev, 2024).

## References

Biswas D (2023) *Contextualizing large language models (LLMs) with enterprise data.* LinkedIn Pulse. Available at: https://www.linkedin.com/pulse/contextualizing-large-language-models-llms-enterprise-debmalya-biswas/ [Accessed 16 July 2023].

D'Amico RD, Erkoyuncu JA, Addepalli S & Penver S (2022) *Cognitive digital twin: An approach to improve the maintenance management.* CIRP Journal of Manufacturing Science and Technology 38: 123–134.

El Mokhtari K, Panushev I & McArthur JJ (2022) *Development of a cognitive digital twin for building management and operations.* Frontiers in Built Environment 8. doi:10.3389/fbuil.2022.856873.

Fernandez T (2023) *How to choose the best OpenAI model for your AI application.* Semaphore CI. Available at: https://semaphoreci.com/blog/openai-models [Accessed 10 August 2023].

Lin KL & Suen JB (2019) *A vision-based method for determining degradation level of a road marking.* Presented at the ATINER Conference Presentation Series No. CIV2019-0138, Athens, Greece.

Mateev M (2023) *Design and implementation of cognitive digital twins with generative AI and ChatGPT.* Presented at the 4th Annual International Conference on Computer and Software Engineering (ATINER SFW2023-0344), Athens, Greece, 17–20 July 2023.

Mateev M (2023) *Predictive analytics based on digital twins, generative AI, and ChatGPT.* Presented at the 27th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI 2023), pp. 168–174. doi:10.54808/WMSCI2023.01.168.

Mateev M (2023) *Using digital twins, IoT, and anomaly detection for predictive analysis in construction industry.* Presented at the 19th Annual International Conference on Information Technology & Computer Science, Athens, Greece, 15–18 May 2023, Abstract Book, p. 68.

Mateev M (2024) *Evolution of predictive analysis using GPT OpenAI models.* Presented at Industry 4.0 2024 Winter Session Proceedings, Scientific-Technical Union of Mechanical Engineering "Industry 4.0", Borovets, Bulgaria, 11–14 December 2024, vol. 2, pp. 78–81.

Mateev M (2024) *Implementing hybrid (AI and data analytics) solutions for optimal performance and cost optimization for image analysis with GPT-4 Turbo with Vision for predictive analysis.* Presented at World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI 2024), pp. 74–80. doi:10.54808/wmsci202.[01.74.

Mateev MA (2025) *Comparative analysis on implementing embeddings for image analysis.* Journal of Information Systems Engineering and Management 10(17s): 89–102. doi: 10.52783/jisem.v10i17s.2710

Ramaji I (2024) *Advancing building management: Digital twins for sustainable HVAC efficiency.* Presented at the 12th Annual International Conference on Industrial, Systems & Design Engineering, Athens, Greece, 24–27 June 2024, Abstract Book

Ünal P (2023) *Cognitive digital twins: Digital twins that learn by themselves, foresee the future, and act accordingly.* Digital Twin Consortium. Available at: https://www.digitaltwinconsortium.org/2022/09/cognitive-digital-twins-digital-twins-that-learn-by-themselves-foresee-the-future-and-act-accordingly/ [Accessed 9 June 2023]